



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

ÚSTAV AUTOMATIZACE A INFORMATIKY

INSTITUTE OF AUTOMATION AND COMPUTER SCIENCE

VYUŽITÍ METOD UMĚLÉ INTELIGENCE PRO SIMULACI A IDENTIFIKACI DAT V OBLASTI PROUDĚNÍ

UTILIZATION OF ARTIFICIAL INTELLIGENCE FOR SIMULATION AND DATA IDENTIFICATION IN
THE FIELD OF FLOWING

DIZERTAČNÍ PRÁCE

DOCTORAL THESIS

AUTOR PRÁCE

AUTHOR

Ing. Jan Richter

ŠKOLITEL

SUPERVISOR

prof. RNDr. Ing. Jiří Šťastný, CSc.

BRNO 2018

Abstrakt

Proudění vzduchu lze vizualizovat pomocí aditiv tak, aby bylo možné pořídit snímky a videozáznamy takového proudění. Aditiva mohou být ve formě částic, nebo tvoří spojitá vlákna. Počítačové zpracování takových dat se liší právě podle použité vizualizační metody. Tato práce pojednává o množství postupů k určení tvaru proudnice a rychlosti proudění ve snímcích a záznamech proudění vzduchu. K těmto účelům jsou využívány kromě exaktních postupů i neuronové sítě a genetické algoritmy.

Abstract

It is possible to simulate an airflow by additives to shoot images and records of such flowing. Additives can be in the form of particles or continuous filaments. A computer evaluation of such data differs depending on the kind of visualization method. This thesis deals with a number of different approaches to determine the airjet shape and airflow velocity in airflow images and records. Exact procedures are used for these purposes as well as neural networks and genetic algorithms.

Klíčová slova

Vícevrstvá perceptronová neuronová síť, genetický algoritmus, zpracování obrazu, vizualizace proudění vzduchu, rychlost proudění, tvar proudnice

Keywords

Multilayer perceptron neural network, genetic algorithm, image processing, visualization of airflow, velocity of airflow, airjet shape

RICHTER, J. *Využití metod umělé inteligence pro simulaci a identifikaci dat v oblasti proudění*. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2018. 120 s. Vedoucí dizertační práce prof. RNDr. Ing. Jiří Šťastný, CSc.

PROHLÁŠENÍ

Prohlašuji, že jsem dizertační práci *Využití metod umělé inteligence pro simulaci a identifikaci dat v oblasti proudění* vypracoval samostatně pod vedením svého školitele prof. RNDr. Ing. Jiřího Šťastného, CSc. s pomocí informačních zdrojů uvedených v seznamu literatury.

.....
Ing. Jan Richter

Děkuji na tomto místě svému školiteli a vedoucímu dizertační práce prof. RNDr. Ing. Jiřímu Šťastnému, CSc. za cenné odborné rady a vedení práce. Dále děkuji své rodině za podporu a motivaci ke studiu. Děkuji také kolegům z energetického ústavu za poskytnutí snímků a záznamů proudění vzduchu, i za cenné rady a připomínky.

Ing. Jan Richter

OBSAH

1	ÚVOD	12
1.1	Cíl práce	12
1.2	Členění práce	13
2	VYHODNOCOVÁNÍ PROUDĚNÍ VZDUCHU	14
2.1	Sledované vlastnosti proudícího vzduchu.....	14
2.1.1	Tvar oblasti zaváděného vzduchu.....	14
2.1.2	Rychlost proudění.....	14
2.1.3	Další vlastnosti proudícího vzduchu	14
2.2	Způsoby vyhodnocování proudění vzduchu.	14
2.2.1	Měření vlastností vzduchu.....	15
2.2.2	Optické metody pro určení vlastností vzduchu	15
2.2.3	Vizualizační metody pro určení vlastností vzduchu.....	15
2.3	Typy snímků proudění vzduchu.....	16
3	METODIKA A TEORETICKÉ PODKLADY	19
3.1	Reprezentace obrazu	19
3.2	Preprocessing	20
3.2.1	Úpravy geometrie	20
3.2.2	Úpravy barev	20
3.3	Hranová detekce.....	21
3.4	Zpracování sekvence snímků.....	21
3.5	Neuronové sítě	22
3.6	Genetické algoritmy.....	24
3.6.1	Velikost populace	25
3.6.2	Výběr rodičů	25
3.6.3	Parametry obměny populace	26
3.6.4	Způsob křížení	27
3.6.5	Mutace	27
3.6.6	Konstrukce potomka z chromozomu.....	28
3.6.7	Ukončení algoritmu	28
4	NOVĚ VYVINUTÉ METODY PREPROCESSINGU	29
4.1	Nelineární hranový detektor	29
4.2	Hranová detekce použitím neuronových sítí.....	32
5	URČENÍ TVARU PROUDU VIZUALIZOVANÉHO MLHOU	34

5.1	Jednoduchá evaluace obrazu.....	34
5.1.1	Prahování.....	34
5.1.2	Podmíněné prahování.....	34
5.1.3	Vyhlažovací algoritmus.....	36
5.2	Vyšetřování proudnice v řezech.....	38
5.2.1	Vyhodnocení řezu.....	39
5.2.2	Sestrojení hranice proudu.....	41
5.3	Rozpoznávání mlhy neuronovými sítěmi.....	42
5.3.1	Princip vyhodnocení neuronovou sítí.....	42
5.3.2	Základní zpracování snímku s využitím neuronové sítě.....	43
5.3.3	Podmíněné prahování spojitě detekce.....	47
5.3.4	Zvětšení neuronové sítě.....	49
5.3.5	Metoda dvou expertů.....	50
5.3.6	Podmíněná detekce dvojicí expertů.....	52
5.3.7	Další využití detekce neuronovými sítěmi.....	55
6	URČENÍ TVARU PROUDU VIZUALIZOVANÉHO HELIOVÝMI BUBLINAMI.....	57
6.1	Detekce heliových bublin náležících proudu.....	57
6.1.1	Určení pixelů náležících heliovým bublinám.....	57
6.1.2	Výběr reprezentantů heliových bublin.....	59
6.1.3	Separace neplatných reprezentantů.....	61
6.1.4	Seřazení množiny reprezentantů.....	63
6.2	Určení konvexní obálky proudnice.....	63
6.3	Nekonvexní obálka proudnice.....	65
6.3.1	Použití genetického algoritmu při konstrukci nekonvexní obálky.....	66
6.3.2	Sestrojení hranice dle chromozomu.....	68
6.3.3	Fitness funkce.....	72
6.3.4	Průběh genetického algoritmu.....	75
7	URČENÍ RYCHLOSTI PROUDU VIZUALIZOVANÉHO MLHOU.....	78
7.1	Princip použití vzájemné korelace.....	78
7.2	Vektorová mapa.....	80
7.3	Další možnosti určení rychlosti.....	81
8	URČENÍ RYCHLOSTI PROUDU VIZUALIZOVANÉHO HELIOVÝMI BUBLINAMI.....	82
8.1	Sledování částic.....	82
8.1.1	Inicializace procesu.....	82
8.1.2	Generování možných cest.....	83

8.1.3	Rozvíjení cesty	83
8.1.4	Ohodnocení cest	85
8.2	Zpracování detekovaných vektorů	88
8.2.1	Porovnávání okolních vektorů	88
8.2.2	Mazání kolizních vektorů	89
8.3	Vektorová mapa	90
8.3.1	Výběr vektorů k učení	90
8.3.2	Nastavení neuronové sítě k určení rychlosti v konkrétním bodě	93
8.3.3	Tvorba vektorové mapy	95
9	URČENÍ VLASTNOSTÍ VENTILAČNÍHO SYSTÉMU	97
9.1	Vyšetřování tvaru proudnice v jediném snímku	97
9.1.1	Určení dosahu proudu a hraničních křivek	97
9.1.2	Určení polohy vyústky	98
9.1.3	Výběr reprezentantů hranice	99
9.1.4	Hraniční přímky, osa proudu a rozptyl	99
9.2	Vyšetřování tvaru proudnice v sekvenci	100
9.2.1	Statistické zpracování výsledků jednotlivých detekcí	100
9.2.2	Sčítání detekovaných pixelů	101
9.3	Vyhodnocení rychlosti	103
9.3.1	Validace vektorové mapy	104
9.3.2	Filtrace vektorové mapy	104
9.3.3	Zpracování rychlosti v sekvenci snímků	104
10	ZHODNOCENÍ A VÝSLEDKY PRÁCE	106
10.1	Tvar proudnice	106
10.2	Rychlosti proudění	108
11	ZÁVĚR	110
11.1	Vědecký přínos práce	110
11.2	Praktický přínos práce	111
12	SEZNAM POUŽITÉ LITERATURY	112
13	SEZNAM POUŽITÝCH ZKRATEK A SYMBOLŮ	115
14	SEZNAM OBRÁZKŮ	116
15	SEZNAM TABULEK	119
16	SEZNAM PŘÍLOH	120

1 ÚVOD

Prakticky každý dopravní prostředek dnes obsahuje systém cirkulace vzduchu. Někdy se jedná pouze o ventilační a výhřevné systémy, ve většině případů se k nim přidává i možnost chlazení. Klimatizace jsou dnes běžně instalovány i v budovách. Samozřejmě se staly ve veřejných prostorech, jako jsou obchody, kina či úřady. Stále častěji se objevují i v rodinných domech.

Efektivita ventilačních a klimatizačních systémů je značně závislá nejen na výkonnosti chladicích či výhřevných zařízení, ale i na jejich vhodném nastavení s ohledem na to, jak se distribuovaný vzduch šíří chlazeným či vytápěným prostorem. Vhodným návrhem ventilačního systému můžeme nejen zvýšit komfort osob v prostoru, ale také snížit energetické nároky takové ventilace. Při návrhu a nastavení ventilačního systému je velmi důležité mapovat chování vzduchu v prostoru. Takové mapování pak bude sloužit k ověření funkčnosti ventilačního zařízení a pro určení některých jeho vlastností. Povědomí o chování vzduchu v prostoru lze získat například vhodným rozmístěním měřicích zařízení, která budou poskytovat informace o teplotě, tlaku, hustotě vzduchu, o rychlosti nebo směru jeho proudění.

Zcela odlišným přístupem jsou vizualizační či optické metody, které umožňují získat snímky a videozáznamy proudění vzduchu, jež je poté třeba počítačově zpracovat. Principem těchto metod je zviditelnit proudnici vzduchu. V případě optických metod se využívá optických zařízení. V případě vizualizačních metod jsou data snímána běžnými fotoaparáty a kamerami, avšak neprůhledný vzduch je mísen nebo úplně nahrazen viditelnými látkami. Oba principy vedou k získání snímků a záznamů, na nichž je proud vzduchu patrný. Po získání snímků proudění vzduchu nastává úkol tyto snímky vyhodnotit a určit z nich důležité vlastnosti proudícího vzduchu, jako jsou tvar a dosah proudnice, rychlost proudění v konkrétním místě, hustota proudu, lokální teplotní mapa, výskyt turbulencí apod.

Pro tyto účely lze užít exaktních vyhodnocovacích algoritmů, ale také metod umělé inteligence. Tato práce se zabývá vyhodnocováním vybraných typů snímků. Vyhodnocovanými kritérii jsou zejména tvar proudu a rychlost proudění v různých místech prostoru. Pro určení těchto vlastností jsou vhodné snímky vzniklé vizualizačními metodami zavádění látek do proudu vzduchu.

1.1 CÍL PRÁCE

Základním úkolem práce je vylepšit stávající metody či vyvinout zcela nové postupy pro zpracování 2D obrazu, aplikovatelné pro vyhodnocení snímků proudění vzduchu tak, aby dokázaly správně stanovit požadované vlastnosti tohoto proudění. Aby byly tyto postupy univerzální, je velmi výhodné v rámci nich využívat metody umělé inteligence. Cílem je, aby zpracování snímků bylo méně závislé na prostředí, v němž byly tyto snímky pořízeny, jako je tomu v případě, kdy takové vyhodnocování provádí člověk.

Dalším úkolem práce je doplnit stávající možnosti tak, aby byly pro oba typy vizualizace (částicemi i spojitými vlákny) získány kvalitativně stejné výsledky. Z metod umělé inteligence je velmi užitečné využít umělých neuronových sítí, které jsou použitelné jako účinné detektory nebo slouží pro realizaci rozhodovacích procesů. Umělé neuronové sítě jsou právě inspirovány činností lidského mozku. Jejich použití slibuje kvalitativně srovnatelné vyhodnocení s vyhodnocením, jaké vytvoří člověk. Jako optimalizační algoritmus je výhodné využívat evoluční metodu genetických

algoritmů. Genetický algoritmus napodobuje evoluci populace, kdy se z počátečních primitivních jedinců vyvíjejí dokonalejší s ohledem na zvolené hodnotící kritérium.

Komplexním úkolem práce je také vývoj počítačového softwaru na zpracování snímků proudění vzduchu, v němž budou uvažované metody implementovány a pomocí kterého bude ověřována jejich funkčnost. Tento software by měl být dobře využitelný mimo jiné v oborech zabývajících se návrhy a konstrukcemi ventilačních systémů.

1.2 ČLENĚNÍ PRÁCE

Práce se zabývá jednotlivými metodami počítačové analýzy pro vyhodnocování snímků a záznamů proudění vzduchu. Zmiňuje důležité preprocessingové funkce použitelné pro úpravy snímků před vlastními vyhodnocovacími procesy. Tyto funkce dokonce v některých případech dokážou vyhodnocovací procesy zcela nahradit.

Práce se dále soustředí na dva typy vizualizace proudění vzduchu. První zkoumanou vizualizační metodou je mísení proudícího vzduchu s mlhou, druhou je metoda zavádění heliových bublinek do proudu vzduchu. Protože tyto metody produkují odlišné typy výsledných snímků, je třeba také jejich vyhodnocení provádět zcela odlišnými postupy. U obou typů vizualizace se práce soustředí na určení dvou základních veličin proudění, a to tvaru proudnice a rychlosti proudění. Vhodnost použití uvažovaných metod je poté zhodnocena, na základě čehož jsou poskytnuta další doporučení a návody na detekci dalších veličin proudu.

2 VYHODNOCOVÁNÍ PROUDĚNÍ VZDUCHU

Tato práce se zabývá analýzou vzduchu, který proudí z vyústky ventilačního systému do prostoru. Konkrétními příklady mohou být ventilační systém pasivního domu či klimatizace automobilu.

2.1 SLEDOVANÉ VLASTNOSTI PROUDÍCÍHO VZDUCHU

Pro popsání a ověření funkčnosti systému ventilace je důležité určit významné vlastnosti proudění vzduchu.

2.1.1 Tvar oblasti zaváděného vzduchu

Jak uvádí například Janotková (1991), dle tvaru oblasti, určené vzduchem vehnaným do místnosti poblíž vyústky lze určovat charakteristiky vyústky. Pokud bude vehnaný vzduch monitorován i v místech vzdálenějších od vyústění systému, lze z tvaru jím určené oblasti usuzovat na způsob, jakým se vzduch šíří po interiéru a zaplňuje jej. Tvar proudnice je jednou ze dvou hlavních vlastností proudícího vzduchu, kterou má tato práce za cíl detekovat.

2.1.2 Rychlost proudění

Podle rychlosti proudění lze odvodit, jak proudění budou vnímat osoby v místnosti. Pro dobrou tepelnou pohodu je vhodnější, aby osoby nebyly proudění vystaveny přímo, ale aby k nim bylo distribuováno velké množství vzduchu. Faktor rychlosti tedy rovněž slouží k optimalizaci tepelné pohody uvnitř (Lízał, Jedelský, Jícha et al., 2016). Rychlost je druhou hlavní charakteristikou proudění vzduchu a také jí se bude tato práce zabývat.

2.1.3 Další vlastnosti proudícího vzduchu

O dalších vlastnostech proudícího vzduchu hovoří například Cermak (2003). Zmiňuje například **turbulenci proudění**. Tu lze vyhodnocovat pomocí detailní mapy vektorů rychlosti proudění. Turbulence jsou četnější pro vyšší rychlosti a negativně zkreslují detekovaný tvar proudnice.

Ke zjištění efektu, jaký ventilace přináší, může dopomoci i **hustota aditiva**. Tu lze určovat na základě barevného odstínu v daném místě snímku. Hustota aditiva demonstruje efektivitu ventilačního systému v konkrétním bodě. Dle ní lze usuzovat, jaké množství tepla může proud v tomto místě předat či odebrat.

S tvarem proudnice a rychlostí proudění souvisí také **dosah proudu**. Lze říci, že proudnice dosahuje do míst, kde proudění nabývá jisté rychlosti, tedy kde se vzduch ještě nezastavil. Dosah proudu je určován z tvaru proudnice, ale může být určen i z vektorové mapy (viz kapitola 7.2) podle přítomnosti rychlosti proudu.

2.2 ZPŮSOBY VYHODNOCOVÁNÍ PROUDĚNÍ VZDUCHU.

Vlastnosti proudu uvedené v kapitole 2.1 lze sledovat různými přístupy. V této kapitole budou popsány tři významné postupy.

2.2.1 Měření vlastností vzduchu

Jedná se o konvenční přístup vyhodnocování proudění, kdy k jeho realizaci je použito sítě měřicích přístrojů, například pro měření rychlosti anemometry, pro měření tlaku barometry, pro měření teploty termometry apod. Měřicí zařízení bývají rozmístěna v místnosti, a to především v místech, kde se očekává přítomnost osob, nebo v místech jinak důležitých pro komplexní zhodnocení proudění (Pokorný, Poláček, Fojtlín et al., 2016).

Fojtlín a Fišer (2016) popisují pokročilejší techniku mapování vlivu proudění na člověka. Jedná se o tzv. tepelného manekýna. Jde o speciálně navrženou figurínu, která je pokryta senzory tak, aby bylo možné usuzovat na stav, v němž by se na místě manekýna nacházel člověk.

Měřicí zařízení jsou propojena s počítačem, který sbírá jimi naměřené hodnoty. Ty jsou poté zpracovávány matematickými a statistickými postupy.

2.2.2 Optické metody pro určení vlastností vzduchu

Zcela odlišnými metodami vyhodnocování proudění jsou takové, jejichž principem je pořízení snímku či videozáznamu, který vypovídá o proudění vzduchu. Získaný výstup je následně vyhodnocen počítačovými metodami založenými na zpracování obrazu.

Jak uvádí Merzkirch (1987), k získání snímků vypovídajících o způsobu proudění je možné využít optické metody. Ty monitorují optické jevy, které způsobuje právě proudící vzduch. K pořízení snímků či videozáznamů proudění se používá k tomu určené speciální snímací zařízení. Mezi nejznámější optické metody patří například infračervená termografie, jejíž princip popsal fyzik Max Planck. V rámci termografické analýzy je mapováno teplotní pole pomocí termovizních kamer, snímajících pomocný materiál proměnné teplotě.

Hariharan (2007) uvádí další optickou metodu, a to interferometrii. Ta využívá optické zařízení interferometr. V rámci procesu získání snímku je monochromatické (má jedinou vlnovou délku) a koherentní (všechny vlny mají stejnou fázi) světlo laseru rozděleno polopropustným zrcadlem na referenční a objektový světelný svazek, který prochází přes měřený proud. Po opětovném sloučení obou svazků se vlny buď sčítají (světlá) nebo odčítají (tmavá) podle jejich aktuální fáze. Světlé a tmavé oblasti tvoří tzv. interferometrické proužky. Výsledný snímek složený z těchto proužků se nazývá interferogram. Možnosti zpracování interferogramu uvádí například Richter a Šťastný (2011).

2.2.3 Vizualizační metody pro určení vlastností vzduchu

Stejně jako v případě optických metod je principem vizualizačních metod získání snímků proudění a jejich následné počítačové zpracování. V tomto případě je však proudění ve snímku patrné díky tomu, že jsou do proudu jinak bezbarvého vzduchu zaváděny viditelné látky, nebo je vzduch těmito látkami zcela nahrazen. Protože je proudění díky těmto látkám viditelné, pro následné pořízení snímků či videozáznamů proudění lze použít i běžné fotoaparáty a kamery (Janotková, Pavelek a kol., 2010). Zaváděnými látkami mohou být látky tvořící souvislá vlákna. Vhodným typem je například mlha, kouř nebo jinak obarvený vzduch. Výsledné snímky jsou vhodné pro určení tvaru proudnice a způsobu, jakým vzduch zaplňuje interiér. Jiným typem vizualizačních prostředků jsou látky, které netvoří souvislá vlákna ale částice, například jiskry, saze, piliny nebo heliové bublinky. Tyto záznamy jsou výhodné pro určení rychlosti proudění a turbulencí. Proud lze

vizualizovat také vlákna lehkých objektů pevného skupenství, jako jsou pruhy látky nebo nitě. Tvar těchto objektů následně definuje tvar proudnice (Janotková, Pavelek a kol., 2010).

Uvedené typy získávání vlastností vzduchu se liší svojí použitelností. Běžná měření vlastností vzduchu vyžadují hustou síť měřidel. I tak produkují méně informací v porovnání s vizualizačními či optickými metodami, jejichž výsledkem je pro daný okamžik bitmapa. Informace o proudícím vzduchu je obsažena v každém jejím pixelu. Síť informací je tak daleko hustší, než může být síť měřidel.

Optické metody mají oproti vizualizačním tu výhodu, že k pořízení snímku není třeba proudící vzduch mísit s žádnou látkou. Jejich nevýhodou naproti tomu je, že je nutné používat specializovaná snímací zařízení.

Tato práce se soustředí na vizualizační metody vyhodnocování proudícího vzduchu. Bude se věnovat oběma podtypům tohoto přístupu, tedy vizualizaci s využitím látek, která tvoří souvislá vlákna (konkrétně mlha), i vizualizaci s využitím částic (konkrétně heliové bublinky).

2.3 TYPY SNÍMKŮ PROUDĚNÍ VZDUCHU

Ve snímcích proudění vzduchu může být vizualizovaný vzduch velmi dobře viditelný, ale také může splynout s pozadím. Je proto vhodné rozčlenit snímky proudění vzduchu podle jejich složitosti. Pavelek, Janotková a Štětina (2001) používají členění, které bude ctít i tato práce.

Snímky prvního typu

Jedná se o snímky pořízené v laboratorních podmínkách tak, aby na proudění nemělo vliv okolní prostředí. Vzduch, který proudí do měřicího prostoru, je velmi zahuštěn aditivem. Pozadí měřicího prostoru má jednotnou barvu zcela odlišnou od barvy aditiva. Proud rovněž obsahuje jen minimum turbulencí, např. díky nižší rychlosti proudění či díky použití specifických prvků systému ventilace, který proud usměrňuje. Příklad snímku prvního typu je na snímku 1.



Snímek 1 Příklad snímku prvního typu

Zdroj: Odbor termomechaniky a techniky prostředí, VUT FSI [s.a.]

Snímky druhého typu

Na těchto snímcích je vizualizovaný proud stále zřetelný. Snímky však nevznikají v ideálním nastavení. K vizualizaci nemusí být použito husté aditivum. Proudění je často turbulentní. Také prostředí nemusí být tak dobře rozlišitelné od aditiva. Pozadí snímku nemusí být zcela homogenní. Příklad snímku druhého typu je na snímku 2.



Snímek 2 Příklad snímku druhého typu

Zdroj: Odbor termomechaniky a techniky prostředí, VUT FSI [s.a.]

Snímky třetího typu

Tyto snímky sice proudění zachycují, avšak zejména kvůli nevyhovujícímu (většinou heterogennímu) pozadí je velmi obtížné tvar proudnice přesně určit, a to i v případě, že vyhodnocování provádí přímo člověk.



Snímek 3 Příklad snímku třetího typu
Zdroj: Odbor termomechaniky a techniky prostředí, VUT FSI [s.a.]

3 METODIKA A TEORETICKÉ PODKLADY

V této kapitole budou zmapovány vybrané matematické postupy a algoritmy. Zmíněny budou užitečné metody zpracování obrazu. Také je důležité představit některé metody umělé inteligence, které budou využity ve specializovaných postupech pro vyhodnocení snímků proudění vzduchu.

3.1 REPREZENTACE OBRAZU

Jak uvádějí Galer a Horvat (2003) barevný obraz se skládá z barev. Pro reprezentaci barev se používají různé barevné modely, jako například *CMY* (Cyan, Magenta, Yellow), *CMYK* (Cyan, Magenta, Yellow, Black), *HSV* (Hue – odstín, Saturation – sytost, Value – jas) apod. Monitory a projektory používají aditivní způsob mísení barev dle barevného modelu *RGB* (Red – červená, Green – zelená, Blue – modrá), tedy i v práci bude operováno s tímto barevným modelem.

Digitální snímek se skládá z obrazových bodů nazývaných pixely. Pixely jsou nejmenší objekty bitmapy snímku. Všechny pixely mají stejnou velikost, čtvercový tvar a každý má určitý barevný odstín. Množství různých barevných odstínů, kterých může každý pixel nabývat, definuje barevnou hloubku snímku. Například pixel bitmapy s barevnou hloubkou 1 bit může nabývat pouze dvou barev, a to černé nebo bílé.

V práci bude využívána barevná hloubka 24 bitů. Tato barevná hloubka nese název „true color“, a to proto, že pokrývá přibližně takové množství barevných odstínů, jaké je schopen člověk rozlišit. 24 bitů barvy je rozděleno na 3 byty, přičemž každý byte přísluší jedné ze tří barevných složek *RGB*. Barevný model *RGB* bývá někdy rozšířen o dalších 8 bitů, které přísluší tzv. alfa kanálu. Alfa kanál určuje průhlednost pixelu. Tento barevný model se jmenuje *RGBA*, avšak v práci nebude alfa kanál pixelu využíván.

Výraz „hodnota pixelu“ bude představovat hodnotu jeho barevných složek *RGB*, v programovacím jazyce Delphi nesených jedinou celočíselnou hodnotou *P*.

$$P = 2^0 \cdot R + 2^8 \cdot G + 2^{16} \cdot B \quad (3.1)$$

Každá ze složek *R*, *G*, *B* nabývá hodnoty $\langle 0, 2^8-1 \rangle$. Výsledná hodnota *P* je pak celé číslo v rozmezí $\langle 0, 2^{24}-1 \rangle$.

Kromě jednotlivých barevných složek lze barvu pixelu hodnotit také jeho jasem *I*, tedy světelnou intenzitou. Podle Galera a Horvata (2003) jas nehodnotí barevné složky jednotlivě, ale představuje souhrnnou světlost pixelu. Pro účely této práce byl použitý jednoduchý převod:

$$I = \frac{1}{3}R + \frac{1}{3}G + \frac{1}{3}B \quad (3.2)$$

Jas tedy nabývá stejného rozmezí hodnot, jako jednotlivé barevné složky, a to $\langle 0, 255 \rangle$. Jas bývá používán v případě převodu z barevného snímku na snímek ve stupních šedi, kdy se nová hodnota všech tří barevných složek bude rovnat původní (i nové) hodnotě jasu.

Poznámka: Lidské oko vnímá barevné složky *R*, *G*, *B* různě intenzivně (zelenou vnímá jako nejsvětlejší, modrou nejtmavší). Pokud je účel převodu snímku na stupně šedi čistě estetický, je vhodnější jas, a tedy i novou hodnotu barevných složek určit například dle ITU-R BT.709 standardu z roku 2015 (International telecommunication union, ©2008).

$$I = 0.2126 \cdot R + 0.7152 \cdot G + 0.0722 \cdot B \quad (3.3)$$

Snímek je obvykle vyhodnocován tak, že jsou vhodně procházeny jeho pixely a porovnávány či upravovány jejich barevné hloubky.

3.2 PREPROCESSING

Jedná se o širokou škálu metod zpracování obrazu, které snímky upravují do vhodné podoby, aby na ně poté mohly být použity specializované vyhodnocovací procedury. Některé preprocessingové metody jsou dokonce tak efektivní, že při jejich aplikaci na vhodné jednoduché snímky mohou problém zcela vyřešit i bez nutnosti aplikace dalších metod.

3.2.1 Úpravy geometrie

Jedná se o soubor metod, které mají za cíl převod zdrojové bitmapy snímku do bitmapy jiné tak, aby se změnila geometrie dotyčného obrazu. Metody tedy operují s rozmístěním pixelů snímku. Úpravy barev jsou realizovány až jako doprovodný krok, aby byl cílový snímek co nejpodobnější snímku zdrojovému.

Změna rozměrů snímku

Jedná se o zvětšení nebo zmenšení v jedné ze souřadných os či v obou současně. Jednoduché exaktní algoritmy, které toho docilují, jsou například Nearest-neighbor interpolation (Jiang, Wang, Mu, 2015) nebo Bilinear interpolation (Abramowitz, Stegun, 1964).

Rotace

Rotace o úhly ve velikosti násobku $\pi/2$ představuje přemísťování pixelů a nedochází při něm ke ztrátě informace. K rotaci o obecný úhel je třeba použít opět interpolaci.

Další geometrické úpravy

K dalším geometrickým úpravám patří například zkosení, posunutí podoblasti v rámci snímku, překlopení a podobně.

3.2.2 Úpravy barev

Tyto metody nijak nemění geometrii snímku, ale operují s hodnotami *RGB* pixelů, aplikují na ně matematické operace a určují novou barevnou hodnotu pixelů.

Jednoduché barevné úpravy

Jako jednoduché lze označit úpravy, kdy se nová *RGB* hodnota každého pixelu stanoví pouze z jeho původní hodnoty. Klasickými příklady těchto úprav jsou negativ (převod na doplňkové barvy), převod na stupně šedi (výsledné hodnoty všech tří barevných složek pixelu jsou si rovny), změna kontrastu či jasu.

Lineární filtrace obrazu

Podle Gonzalese a Woodse (2008) jsou lineární filtry založeny na použití obrazové masky, která slouží jako činitel stejně velké podoblasti. Intenzita jejího prostředního pixelu je stanovena jako součet všech odpovídajících součinů.

Filtr typu **dolní propust** (matice 3×3 hodnot $1/9$) se používá k odstranění šumu, což však vede k rozostření celého snímku. ke stejnému účelu se používá také filtr Gausův, který negativní jev rozmazání zmírňuje. Filtr typu **horní propust** (nejčastěji typu čtyřkolí či osmikolí) naopak zvýrazňuje hrany a rozdíly mezi pixely, ale spolu s tím také šum snímku. K detekci hran lze použít také **Laplaceův filtr** nebo vertikální či horizontální **Sobelův filtr** (Gonzales, Woods, 2008).

Nelineární filtrace

Nelineární filtry využívají nejčastěji statistických metod aplikovaných na pixely blízkého okolí určovaného bodu. Pro zvětšení tmavých oblastí se používá filtr typu **minimum** (hodnota pixelu bude odpovídat nejnižší hodnotě v jeho okolí), pro zvětšení světlých oblastí filtr typu **maximum**. Menší využití mají filtry typu **modus** a **medián**. Těmito a dalšími možnostmi se ve své publikaci zabývají Khare a Nagwanshi (2012).

Procedurální úpravy obrazu

Jedná se o sofistikované algoritmy, které spočívají v procházení snímku (i mnohonásobném) a sledování konkrétních vlastností. Jedním z užitečných algoritmů je **ekvalizace histogramu**. Tou se dle Bassiou a Kotropoulou (2007) rozumí jasové vyvážení snímku. Používá se pro příliš tmavé či příliš světlé snímky. Hodnota pro modifikaci jasu pak není pro všechny pixely konstantní, ale je odvozená právě z histogramu snímku.

3.3 HRANOVÁ DETEKCE

Jedná se o sofistikované algoritmy, které vedou k detekci hran ve snímku, díky kterým je pak možné detekovat celé objekty ve snímku. Jako jednoduché hranové detektory lze použít některé lineární filtry, jak bylo uvedeno v kapitole 3.2.2 v podkapitole Lineární filtrace obrazu. Jedním z nejpoužívanějších hranových detektorů je **Cannyho hranový detektor**. Canny (1986) popisuje postup hranové detekce ve čtyřech krocích: Gaussova filtrace, určení gradientu pomocí Sobelových filtrů, určení lokálních maxim a eliminace nevýrazných hran. **Derichův hranový detektor** pak zachovává postup Cannyho detektoru, ale využívá optimalizované obrazové filtry (Deriche, 1987).

Hranové detektory jsou vhodné především pro detekci heliových bublin. V případě snímků proudění vizualizovaného mlhou nenachází takové využití, protože přechod mezi mlhou a okolím bývá pozvolný, a proto na něm většinou nebývá hrana detekována.

3.4 ZPRACOVÁNÍ SEKVENCE SNÍMKŮ

Sekvencí snímku se bude pro účely této práce rozumět množina snímků, které zachycují totéž proudění vzduchu. Během snímání je neměnná poloha snímacího zařízení ani vyústky. Podle způsobu jejího pořízení se liší kvalita snímků i možnosti vyhodnocení takové sekvence.

Opakované focení

Pro tuto variantu je typické, že časová známka snímku nebude hrát v dalším vyhodnocení roli. Mezi jednotlivými snímky mohou být libovolně dlouhé časové rozestupy. Každý snímek je pořizován zvlášť a lze tak klást důraz na jejich kvalitu. Pro další vyhodnocování je důležité, aby byly podmínky při pořizování snímků neměnné. V množině snímků pak nebude záležet ani na pořadí jednotlivých snímků. Lze z ní také odstranit snímky horší kvality, či snímky, kde bylo proudění nežádoucím způsobem ovlivněno nějakým vnějším vlivem.

Přístup opakovaného focení je výhodný pro statistické zhodnocení zkoumaného proudění. Pokud je stejný postup vyhodnocení použitý na celou množinu snímků, jeho výsledky lze pak podrobit statistickým metodám, a tím získat výsledky obecnější.

Videozáznam

I na videozáznam lze pohlížet jako na množinu snímků. V tomto případě je časová známka snímků velmi důležitá. Snímky tedy nelze co do kvality optimalizovat jako v případě opakovaného focení. Rovněž nelze ze sekvence jakýkoliv snímek odstranit, aniž by to neovlivnilo účel použití videozáznamu.

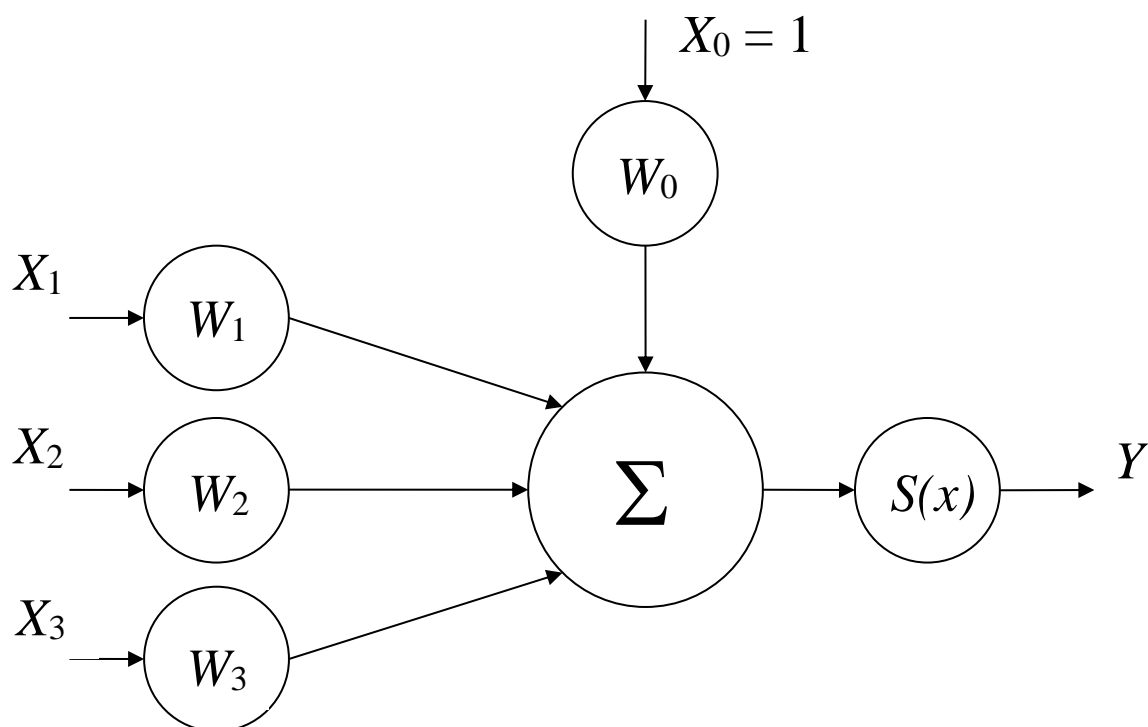
Díky časové známce je znám časový rozdíl mezi jednotlivými snímky. Z obrazového a časového rozdílu mezi snímky lze následně určovat rychlosti proudění. Právě kvůli potřebě určení jeho rychlosti jsou videozáznamy velmi často používány.

Poznámka: Rychlosti lze také určit například prodloužením doby expozice u jednotlivých snímků, avšak takové snímky pak vyžadují speciální sofistikované metody vyhodnocení.

3.5 NEURONOVÉ SÍTĚ

Kromě exaktních metod zpracování obrazu je trendem současné doby používat k tomuto účelu také metody umělé inteligence. Podskupinou těchto metod, hojně využívanou v této práci, jsou umělé neuronové sítě, které mají za úkol simulovat činnost lidského mozku. Podobně jako mozek se i umělá neuronová síť skládá z jednotlivých neuronů a také z komunikačních kanálů (synapsí), kterými si neurony předávají informace.

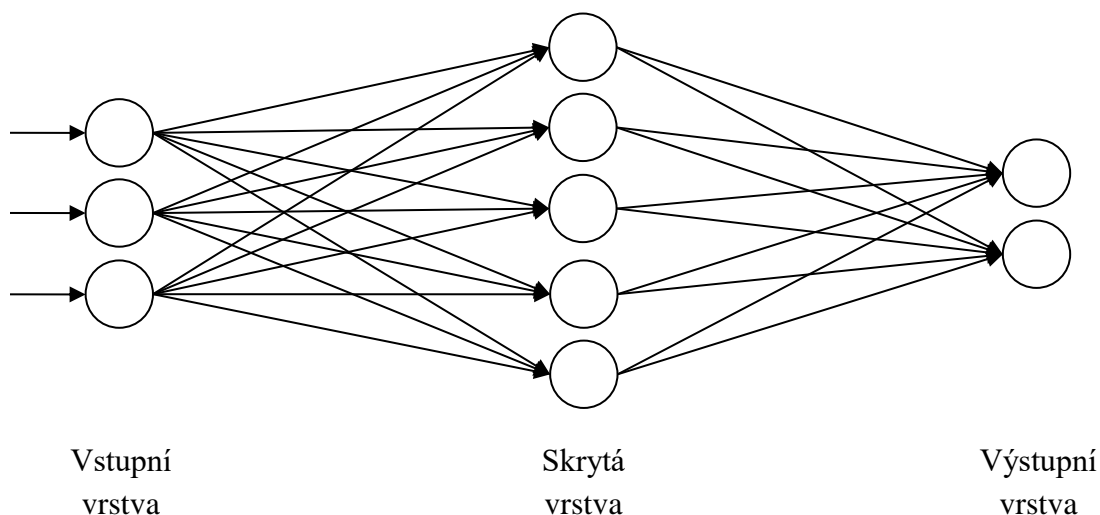
Asi nejpoužívanější model neuronu popsali ve své publikaci McCulloch a Pitts (1943). Podle nich do neuronu vstupují signály pocházející obecně z více neuronů. Ty jsou znásobené jim odpovídajícími nastavitelnými synaptickými vahami, k nimž se připočítává nastavitelný práh. Součet těchto hodnot je vstupem přenosové funkce neuronu, která vyprodukuje jeho jediný výstup. Struktura takového neuronu je patrná ze snímku 4.



Snímek 4 Schéma struktury neuronu navržená McCullochem a Pittsem (1943), kde:

X_1 – X_3 – vstupy neuronu; W_1 – W_3 – synaptické váhy; W_0 – práh; $S(x)$ – přenosová funkce; Y – výstup neuronu

Existuje několik druhů neuronových sítí, avšak v rámci této práce byla použita pouze vícevrstvá dopředná neuronová síť. Tato síť bývá též označována vícevrstvá perceptronová síť, neboť jejím základním prvkem bývá specifický neuron, označovaný jako perceptron. Jak uvádí Hagan (1996), vícevrstvá dopředná síť (multilayer perceptron network, dále jen *MLP*) se skládá z několika vrstev. První vrstva, na níž jsou přiváděna data ke zpracování, se nazývá vstupní vrstva, poslední vrstva, z níž jsou naopak odečítány výsledky, se nazývá vrstva výstupní. Mezi těmito vrstvami může být libovolný počet skrytých vrstev. Synapse mezi neurony v síti jsou realizovány tak, že do každého neuronu vstupují výstupy všech neuronů vrstvy předcházející (s výjimkou vstupní vrstvy). Popsaná struktura *MLP* je schematicky znázorněna na příkladu na snímku 5, kdy má *MLP* tři vstupní neurony, pět neuronů v jediné skryté vrstvě a dva neurony výstupní, proto je značena *MLP* 3-5-2.



Snímek 5 Příklad struktury *MLP* 3-5-2

Vstupy předané vstupním neuronům jsou sítí vyhodnoceny tak, že jsou v dopředném směru po vrstvách vyhodnoceny všechny neurony dle hodnot předchozích neuronů, aktuálně nastavených synaptických vah a aktuálních hodnot prahů. Výsledky sítí vrací jako výstupy neuronů výstupní vrstva. Proces, kdy sítí vyhodnocuje vstupy, je nazýván aktivním režimem sítě.

Aby sítí byla schopná vstupy vyhodnocovat správně, musí mít odpovídajícím způsobem nastaveny zmíněné synaptické váhy a prahy. Nastavení těchto hodnot je nazýváno učením sítě. Hodnoty se nastavují dle předaných vzorů a požadovaných výstupů. Předání vzorů a k nim náležících výsledků provádí tzv. učitel sítě. Samotné učení sítě znamená cyklické provádění úkonu, kdy sítí určí výsledek pro jeden ze vzorů a porovná ho s výsledkem, který stanovil učitel. Pokud dojde k odchylce, bude se šířit neuronovou sítí ve zpětném směru a budou se na ni adaptovat synaptické váhy a prahy jednotlivých neuronů. Tento proces je nazýván adaptivním režimem sítě. Metoda přenastavování prahů a vah se nazývá backpropagation. Po dostatečném počtu cyklů učení je sítí zcela adaptovaná na všechny předané vzory a dokáže replikovat znalosti učitele, které jí byly předány (pokud je architektura sítě vhodně navržena). Sítí je pak schopná správně reagovat i na jiné vstupy, než které jí byly předány k učení. Více o neuronových sítích píše například Šťastný a Škorpil (2007).

Poznámka: Neurony *MLP* sítě využívají jako aktivační funkce nejčastěji sigmoidu.

3.6 GENETICKÉ ALGORITMY

Genetický algoritmus je optimalizační metoda, která slouží k nalezení co nejlepšího řešení daného problému, a to zejména tehdy, pokud pro tuto optimalizaci neexistuje exaktní algoritmus. Genetické algoritmy patří do skupiny evolučních metod, kterým se věnují například Lýsek, Šťastný a Motyčka (2012). Inspirují se právě vývojem populace a evolučními procesy známými z biologie. V obou případech je sledována populace jedinců, kteří si hledají vhodného partnera ke křížení, poté se zkříží a vznikne nový jedinec, který dědí vlastnosti po svých rodičích.

V případě genetických algoritmů je jedincem konkrétní řešení problému. Mitchell (1996) uvádí, že každé řešení je popsáno chromozomem jedince, tedy řetězcem genů. Kvalitu daného řešení určuje

tzv. fitness funkce. V reálném životě lze fitness hodnotu nazývat kvalitou či atraktivitou jedince. Jedná se o ohodnocení toho, jak velkou má jedinec šanci se zkřížit. Hynek (2008) zmiňuje, že jakmile jsou vybráni jedinci ke křížení, dojde k procesu křížení, kdy chromozom potomka vzniká náhodnou kombinací genů chromozomu předků. Výsledný chromozom může poté ještě zmutovat. Nový chromozom pak charakterizuje nového jedince, kterému je rovněž ihned po vytvoření spočtena jeho vlastní fitness hodnota. Tento jedinec se poté zařazuje do populace. Proces umírání je zde realizován coby výměna nového jedince za vybraného staršího.

Koza (1992) vysvětluje, že pokud mají nejkvalitnější jedinci největší šanci se zkřížit, je pravděpodobné, že i jejich potomci budou kvalitní. Úkolem algoritmu je nejenom řídit vývoj populace, ale také si pamatovat aktuálně nejkvalitnějšího jedince. Algoritmus končí po určitém počtu generací, při dosažení jedince určité kvality nebo pokud se po určitý počet generací již nedaří vyšlechtit kvalitnějšího potomka.

Před uvedením genetického algoritmu do chodu je třeba zvolit způsob jeho práce a nastavit jeho parametry, čímž se budou zevrubně zabývat následující podkapitoly.

3.6.1 Velikost populace

Dle Mitchella (1996) se velikost populace odvíjí od délky chromozomu, případně od počtu kombinací genů. Nutnou podmínkou tedy je, aby počet jedinců v populaci nepřevyšoval maximální počet všech kombinací. Protože každý jedinec v populaci má jedinečný chromozom, nemohlo by tak dojít ani k inicializaci populace. Obecně se doporučuje populace v rozmezí 50 – 200 jedinců.

3.6.2 Výběr rodičů

V genetických algoritmech nemusí nutně platit tytéž zákonitosti jako v přírodě. Potomek může mít tedy více než dva rodiče. Teoreticky může mít také pouze jediného a odlišovat se od něj jen díky mutaci. Výběr rodičů pro konkrétní křížení lze dle Mitchella (1996) provádět několika způsoby:

- **Turnajový** – Jedním z rodičů se stane nejkvalitnější jedinec z náhodné neprázdné podmnožiny populace.
- **Ruletový** – Pro každého jedince je stanoven interval $\langle L_{iMin}, L_{iMax} \rangle$, kde $L_{iMin} = L_{(i-1)Max}$ a $L_{iMax} = L_{iMin} + F_i$. F_i je fitness hodnota i -tého jedince. Je vygenerováno náhodné číslo v rozmezí $\langle 0, \sum F \rangle$. Jako rodič bude vybrán ten jedinec, v jehož intervalu $\langle L_{iMin}, L_{iMax} \rangle$ náhodné číslo leží. Každý jedinec má tak šanci $F_i / \sum F$, že bude daným výběrem vybrán.
Poznámka: Při výběru dalšího z rodičů pro konkrétní křížení nelze již vybrané rodiče vybrat znovu. Je proto nutné aktualizovat intervaly i hodnotu $\sum F$. Rychlejší způsob je generovat náhodná čísla, dokud nebude vybrán jiný rodič, než který už byl pro dotyčné křížení vybrán.
- **Proporcionální** – Jedná se o ruletový způsob, do něhož však místo fitness hodnot F_i vstupují upravené fitness hodnoty F'_i . Ty jsou odvozené od zvolené konstanty r . Tato konstanta určuje, kolikrát větší pravděpodobnost být vybrán má nejlepší jedinec oproti nejhoršímu. Hodnotu F'_i vypočítáme jako:

$$F'_i = k \cdot F_i + q \quad (3.4)$$

Kde se konstanty k a q vypočítají ze zvolené konstanty r :

$$k = \frac{r - 1}{F_{max} - F_{min}} \quad (3.5)$$

$$q = \frac{F_{max} - r \cdot F_{min}}{F_{max} - F_{min}} \quad (3.6)$$

Například pokud existují tři fitness hodnoty $F_0 = 4$, $F_1 = 2$ a $F_2 = 8$ a zvolená konstanta $r = 2$, pak

$$k = \frac{2 - 1}{8 - 2} = \frac{1}{6} \quad (3.7)$$

$$q = \frac{8 - 2 \cdot 2}{8 - 2} = \frac{2}{3} \quad (3.8)$$

$$F'_0 = 4 \cdot \frac{1}{6} + \frac{2}{3} = 1.\bar{3} \quad (3.9)$$

$$F'_1 = 2 \cdot \frac{1}{6} + \frac{2}{3} = 1 \quad (3.10)$$

$$F'_2 = 8 \cdot \frac{1}{6} + \frac{2}{3} = 2 \quad (3.11)$$

Tedy platí, že jedinec s nejvyšší fitness hodnotou bude mít v ruletovém výběru potomků r -násobnou šanci být vybrán oproti jedinci s nejnižší fitness hodnotou.

3.6.3 Parametry obměny populace

Jak uvádí Hynek (2008), genetický algoritmus pracuje v krocích. V každém kroku dojde k obměně populace. Obecně lze v rámci této obměny novým jedincem nahradit nejmeně jednoho jedince stávající populace, nejvýše lze nahradit celou populaci (tzv. generační výměna). Nahrazení celé populace novou generací jedinců může být rozporuplné, protože tím zanikne i aktuálně nejlepší jedinec v populaci a není zaručeno, že bude v nové generaci získán ještě lepší. Proto je vhodné si nejlepšího potomka pamatovat, i když již nefiguruje v populaci, dokud není nalezen ještě lepší.

Obecně platí, že pokud se generační obměna bude týkat pouze malého množství jedinců, v procesu výběru rodičů budou opakovaně vybíráni titíž jedinci, tedy populace se nebude vyvíjet. Naopak pokud se obměna bude týkat velkého počtu jedinců, je riziko, že populace přijde o nejkvalitnější jedince s potenciálem dalších velmi kvalitních potomků.

S tímto bodem souvisí i výběr starých jedinců, kteří budou z populace vyloučeni, aby její velikost zůstala po přidání nových jedinců stejná. Pokud se jedná o výměnu celé populace, pak tento výběr nemá smysl. V ostatních případech bude z populace vyloučen adekvátní počet jedinců s nejnižšími fitness hodnotami. Další možností je vybrat jedince k vyloučení náhodně, čímž však roste riziko, že bude vyloučen kvalitní jedinec namísto nekvalitního, kterému bude dána další možnost šířit svoji genetickou informaci.

3.6.4 Způsob křížení

Křížení dle Hynka (2008) představuje konstrukci nového chromozomu podle chromozomů vybraných rodičů. Pro následující příklad bude počet rodičů reprezentovat hodnota m a počet genů hodnota n (pro účely této práce bude konstantní pro všechny jedince). Nový chromozom je náhodně rozdělen na m dílů výběrem $m-1$ hraničních indexů – bodů. Každý z rodičů pak naplní jeden z dílů příslušnými svými geny, jak je znázorněno na příkladu v tabulce 1, kdy dochází ke křížení tří rodičů. Zvolenými hraničními geny jsou 3 a 7.

Tabulka 1 Křížení náhodně dlouhými úseky chromozomu

Index genu	0	1	2	3	4	5	6	7	8
Chromozom 1. rodiče	0	1	2	3	4	5	6	7	8
Chromozom 2. rodiče	10	11	12	13	14	15	16	17	18
Chromozom 3. rodiče	20	21	22	23	24	25	26	27	28
Chromozom potomka	0	1	2	13	14	15	16	27	28

Počet rozdělovacích bodů i výběr rodiče jako zdroje genů pro vyplnění konkrétní části chromozomu lze randomizovat. Limitním případem je, pokud je pro každý gen náhodně volen rodič, z něhož bude naplněn. Příklad takového křížení ukazuje tabulka 2. Pro každý gen byl náhodně vybrán jeden ze tří rodičů, náhodná funkce vrátila postupně tyto indexy: 1, 3, 3, 2, 1, 2, 1, 3, 3.

Tabulka 2 Křížení náhodnými rodiči pro každý gen

Index genu	0	1	2	3	4	5	6	7	8
Chromozom 1. rodiče	0	1	2	3	4	5	6	7	8
Chromozom 2. rodiče	10	11	12	13	14	15	16	17	18
Chromozom 3. rodiče	20	21	22	23	24	25	26	27	28
Chromozom potomka	0	21	22	13	4	15	6	27	28

3.6.5 Mutace

Dle Mitchella (1996) se mutací rozumí náhodná změna náhodného (většinou malého) počtu genů. Chromozom potomka tak může obsahovat i genetickou informaci, kterou neobsahoval žádný z jeho předků. Pokud by se stalo, že se jistá vlastnost (hodnota konkrétního genu) již v populaci nevyskytuje, díky mutaci může vzniknout jedinec, který takovou vlastnost přesto bude mít. Jedná se tedy o pojistku proti ztrátě informace. Mutace může vývoj populace zpomalit, ale zároveň může zamezit stagnaci vývoje, jestliže populace ztratí některé genetické informace.

3.6.6 Konstrukce potomka z chromozomu

Po křížení a mutaci je chromozom nového jedince dokončený. V tento moment je vhodné zkontrolovat, že se totožný chromozom již nevyskytuje v současné populaci, ani mezi již vytvořenými potomky v této populační obměně. Pokud je chromozom unikátní, nastává úkol dle chromozomu jedince vytvořit a zhodnotit. Proces konstrukce jedince z chromozomu se nazývá genotyp, přičemž vytvořený jedinec se nazývá fenotyp. Vytvoření potomka znamená převedení chromozomu na odpovídající řešení problému. Jakmile je řešení kompletní, je třeba na ně aplikovat fitness funkci, díky které získáme jeho fitness hodnotu (Hynek, 2008).

Oba zmíněné úkoly jsou velmi specifické pro každý konkrétní problém, který má genetický algoritmus řešit. Mohou být časově nejnáročnější částí procesu obměny populace. Jsou tak klíčovou otázkou při snaze využít k optimalizaci řešení genetické algoritmy.

Po ohodnocení je nový jedinec připravený zařadit se do populace, jakmile se v ní uvolní místo (viz kapitola 3.6.3). Je to také vhodný okamžik posoudit, zda dotýčný jedinec není ještě kvalitnější než doposud nejkvalitnější nalezený.

3.6.7 Ukončení algoritmu

Hynek (2008) se dále věnuje důležitému rozhodnutí, zda se populace již dostatečně dlouho vyvíjela a zda je nejlepší nalezené řešení dostatečně kvalitní. Práce genetického algoritmu se pak může ukončit. Je několik způsobů, jak genetický algoritmus ukončit:

- **Maximální počet generačních obměn** – Je dopředu stanoveno, kolika generačními obměnami populace projde. Hodnota však musí být volena s ohledem na počet nových jedinců v každé populační obměně. Více vypovídající může být hodnota maximálního počtu všech křížení, avšak pak hrozí, že tato hodnota bude dosažena v průběhu populační obměny.
- **Požadovaná hodnota fitness funkce** – Po každé generační obměně je kontrolováno, zda nejlepší jedinec nepřekračuje zmíněnou požadovanou hodnotu. Pokud ano, algoritmus končí. Typicky se toto kritérium volí v případech, kde má řešení splňovat nějakou procentuální úspěšnost (např. 90 %). Pokud je obtížné určit maximální hodnotu fitness funkce, je i velmi obtížné stanovit hodnotu pro dostatečně kvalitní řešení.
- **Zastavení vývoje** – Populace genetického algoritmu se obecně vyvíjí k čím dál lepším řešením daného problému. Pokud však po jistý počet generačních obměn nedojde ke zlepšení řešení (k nalezení kvalitnějšího jedince, než je aktuálně nejlepší). Lze říci, že tato možnost daleko lépe ošetřuje, aby se algoritmus nezastavil předčasně, ale také aby algoritmus nepracoval příliš dlouho. Jeho nevýhodou je, že nelze dopředu odhadnout, jak dlouho algoritmus bude pracovat.

4 NOVĚ VYVINUTÉ METODY PREPROCESSINGU

Kromě zaměření na snímky proudění vzduchu byly v rámci výzkumu vyvinuty i algoritmy, použitelné pro snímky obecně, bez ohledu na to, co znázorňují.

4.1 NELINEÁRNÍ HRANOVÝ DETEKTOR

Hranové detektory jsou velmi využitelné nejen pro snímky proudění vzduchu. Pro zvýraznění hran lze použít například jednoduché lineární filtry jako filtry typu horní propust, Laplaceův filtr či Sobelovy filtry. Pro jednoznačné označení hran jsou určeny Cannyho či Dierichův hranový detektor.

Specifických výsledků lze však dosáhnout i nelineárním zpracováním okolí pixelů. Nelineární hranový detektor (dále *NHD*) existuje ve dvou variantách, a to *NHD* typu minimum či *NHD* typu maximum, podle toho, ze kterého nelineárního filtru detektor vychází. Princip *NHD* je popsán na hranovém detektoru typu minimum. Prvním krokem je vlastní detekce hranic. V nejbližším okolí $[P_{x-1, y-1} \dots P_{x+1, y+1}]$ vyšetřovaného pixelu P_{xy} je vyhledána nejnížší hodnota každé z barevných složek R , G , B . Dále jsou spočteny rozdíly světelných složek aktuálního pixelu a nalezených minimálních světelných složek v okolí:

$$dR = R_p - R_{min} \quad (4.1)$$

$$dG = G_p - G_{min} \quad (4.2)$$

$$dB = B_p - B_{min} \quad (4.3)$$

Nejvyšší z těchto rozdílů pak bude představovat základ Z_{xy} pro určení nové hodnoty světelné intenzity pixelu P_{xy} . Spolu s výpočty základů pro všechny pixely také vyhledáme nejvyšší základ Z_{max} . Pokud je $Z_{max} = 0$, je zřejmé, že nikde ve snímku neexistuje rozdíl mezi dvěma pixely, tedy zdrojový snímek obsahoval konstantní barvu. V takovém případě samozřejmě nikde ve snímku neexistuje hrana. Každý pixel snímku nabude hodnoty $P_{xy} = 0$.

Pro $Z_{max} > 0$ má smysl pokračovat v detekci hranic a procedura pokračuje vyvažovacím krokem. Vyvažování umožní, aby detektor produkoval kvalitativně shodné výsledky pro snímky jasově vyvážené i nevyvážené. Pokud je hodnota Z_{max} blízká 0, jsou všechny základy pixelů nezřetelné. ve fázi vyvažování produkujeme ze základů pixelů vyvážené intenzity pixelů V_{xy} . Vyvážení představuje uzpůsobení zřetelnosti hran dle zadaného koeficientu k_v , který slouží ke stanovení činitele n_v tak, aby po vyvážení nabyla hodnota vyvážené intenzity

$$V_{max} = I_{max} \cdot k_v = n_v \cdot Z_{max} \quad (4.4)$$

=>

$$n_v = \frac{I_{max} \cdot k_v}{Z_{max}} \quad (4.5)$$

kde I_{max} je hodnota nevyšší možné intenzity.

Vyvážené intenzity pixelů se pak rovnají

$$V_{xy} = Z_{xy} \cdot n_v \quad (4.6)$$

Výsledná intenzita pixelu je pak rovna menší z hodnot V_{xy} a I_{max} . Tedy pro $k_v \geq 1$ bude nejzřetelnější hrana označena jako bílá. Pro $k_v = 1$ nedochází ke ztrátě informace. Přesto pro zřetelnější vyznačení hran může být výhodné volit tento koeficient větší. Pro koeficient $k_v = 2$ budou jako jasné bílé znázorněny všechny hrany s alespoň polovičním základem Z_{xy} oproti Z_{max} .

Poznámka.: Aby algoritmus produkoval ekvivalentní výsledky jako jiné typy hranových detektorů, tedy plnil pixely jen dvěma možnými barvami (bílá pro hranu, černá pro pixel mimo hranu), lze na něj navázat například prahováním (což však nezaručí spojitost hran).

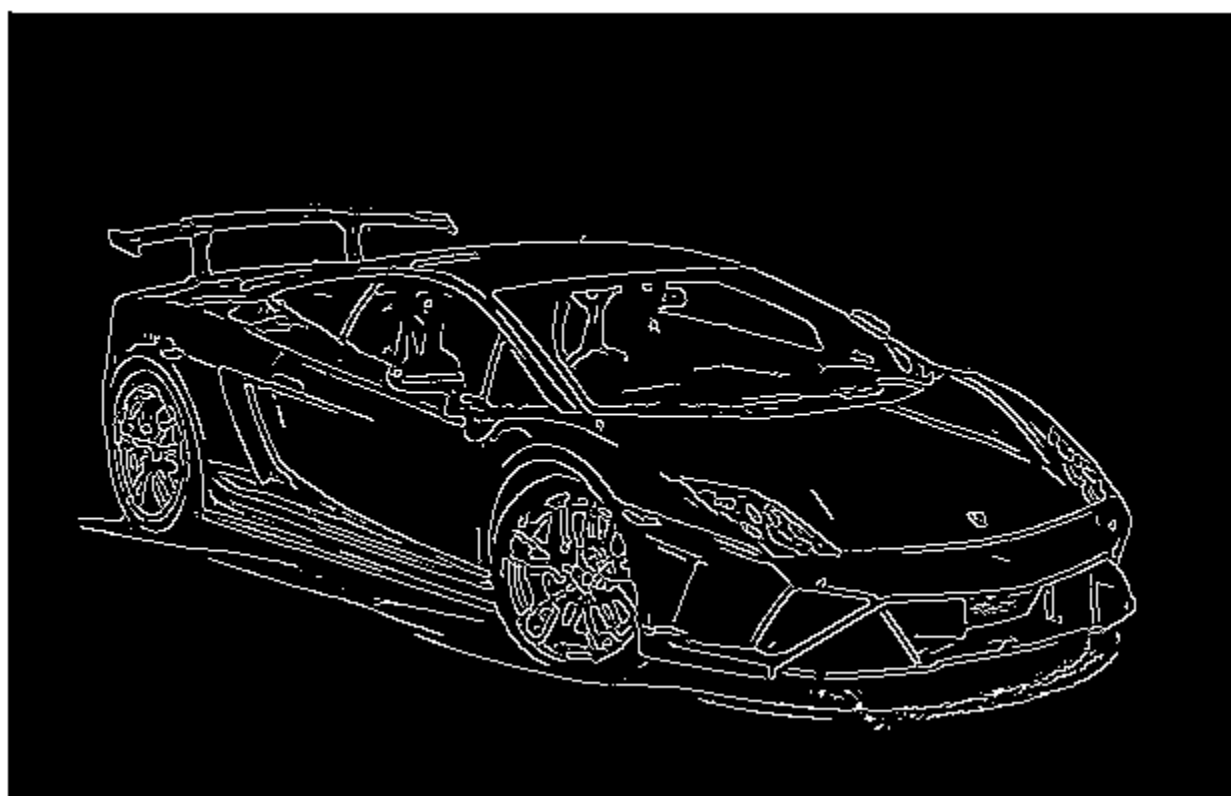
Příklad detekce hran je na následujících snímcích. Na snímku 6 je zdrojový obrázek. Detekce hran pomocí *NHD* je pak na snímku 7 a na snímku 8 je výsledek hranové detekce téhož obrázku Cannyho detektorem.



Snímek 6 Obrázek pro detekci hran
Zdroj: Auto.cz, 2013



Snímek 7 Výsledek nelineární hranové detekce bez prahování, $k_v = 2$



Snímek 8 Výsledek hranové detekce Cannyho hranovým detektorem

Oproti Canny detektoru je *NHD* daleko rychlejší a jeho výsledek se mnohem více blíží způsobu, jakým kreslí člověk rukou. Nevýhodou je, že *NHD* postupuje po bodech a nijak nevyhodnocuje křivky, což vede například k tomu, že může produkovat i relativně široké linky. Po prosté prahování také neošetří spojitost hran. V úvodu této kapitoly bylo zmíněno, že *NHD* existuje ve variantě typu minimum i maximum. Odlišnost *NHD* typu maximum oproti popsanému *NHD* typu minimum je v tom, že základy Z_{xy} jsou počítány jako největší z rozdílů intenzit aktuálních barevných složek a největších hodnot barevných složek v okolí, tedy

$$Z_P = \max(R_{max} - R_P; G_{max} - G_P; B_{max} - B_P) \quad (4.7)$$

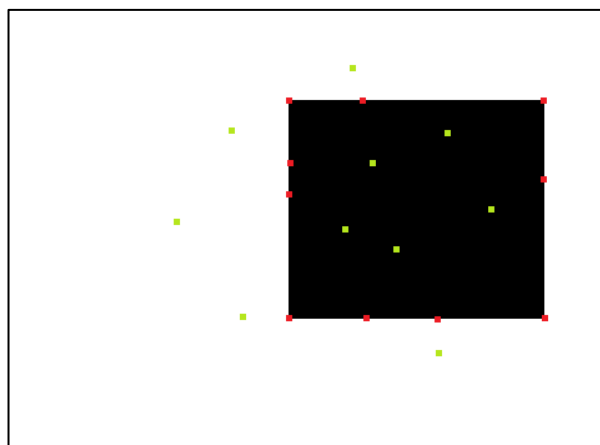
Efekt obou přístupů se liší tím, že zatímco *NHD* typu minimum vykresluje hranu na pozici světlých pixelů a ohraničuje tmavé, *NHD* typu maximum naopak vykresluje hranu na pozici tmavých pixelů a světlé ohraničuje. Například černý pixel na světlém pozadí bude *NHD* typu maximum zvýrazněn, kdežto *NHD* typu minimum jej ohraničí.

4.2 HRANOVÁ DETEKCE POUŽITÍM NEURONOVÝCH SÍTÍ

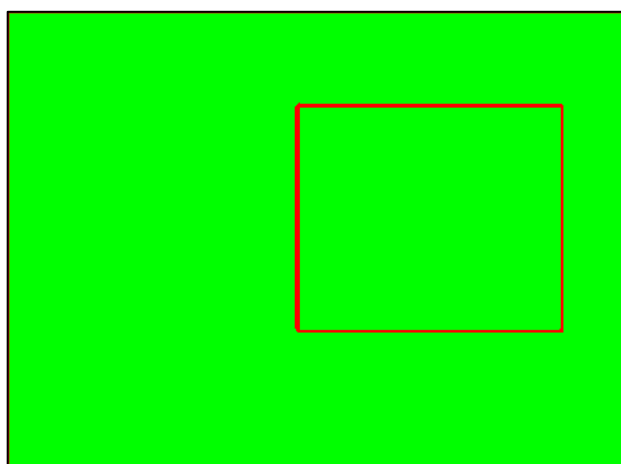
V kapitole 5.3 bude detailně vysvětlen proces prohledávání snímku pomocí neuronových sítí. Stejný princip je však použitelný už v preprocessingové části, konkrétně v určování hran objektů ve snímku. Názorná ukázka je na jednoduchém příkladu na snímku 9. Snímek 9a je zdrojovým obrázkem, v němž mají být detekovány hrany. Neuronové síti jsou předány vzory, jak je znázorněno na snímku 9b (červené oblasti náleží pixelům true, zelené pixelům false). Na snímku 9c je pak dotyčný obrázek zpracovaný sítí typu multilayer feedforward neural network, která se zmíněné vzory naučila.



a) zdrojový snímek



b) předání vzorů



c) výsledek hranové detekce

Snímek 9 Hranová detekce neuronovou sítí

Jak už bylo řečeno, princip celého vyhodnocovacího procesu bude detailně okomentován v kapitole 5.3.

5 URČENÍ TVARU PROUDU VIZUALIZOVANÉHO MLHOU

V následující kapitole budou všechny zmíněné metody věnované vyhodnocování snímků, které zachycují proudění, vizualizované mlhou. Účelem metod bude detekce tvaru proudnice.

5.1 JEDNODUCHÁ EVALUACE OBRAZU

Cílem jednoduchého ohodnocení snímku proudění je získat v krátkém čase přibližné ohraničení mlhy. Proces se skládá z dvojice kroků. Prvním detekujeme pixely, které podle zvoleného kritéria představují mlhu. V dalším kroku je ohraničením těchto pixelů sestrojena hranice proudnice.

5.1.1 Prahování

Pravděpodobně nejjednodušší funkcí, kterou lze použít k účelu detekce pixelů mlhy, je prahování. Pro svoji jednoduchost bývá řazeno do preprocessingových metod. Jak uvádí Galer a Horvat (2003), prahování se nejčastěji provádí dle jasů pixelu, ale lze prahovat třeba jen dle jedné barevné složky. Taková operace má smysl například při vizualizaci proudů použitím barevného aditiva. Sledování jediné barevné složky může být také užitečné při řešení snímků složitějších, než jsou snímky prvního typu.

Prahování obecně spočívá v operaci, kdy pixely, jejichž jas (či konkrétní barevná složka) dosáhne hranice prahu, budou označeny definovanou hodnotou. Pixely, které hranice naopak nedosáhnou, budou označeny hodnotou jinou. Například světlé pixely s jasně alespoň 128 budou označeny bílou barvou, pixely pod touto hranicí budou označeny černou.

Práh se nejjednodušeji stanovuje odhadem dle jasů vizualizačního aditiva a dle jasů pozadí. Užitečnou operací před vlastním prahováním je ekvalizace, která snímek jasově vyváží. Pochopitelně s tím vzroste čas celého procesu, a to na víc než dvojnásobek, protože ekvalizace je sama o sobě časově náročnější operací oproti prahování. Prahování snímků proudění vizualizovaného mlhou se ve své publikaci věnuje Škorpil, Richter a Šťastný et al. (2012).

5.1.2 Podmíněné prahování

Jedná se o nově vyvinuté vylepšení běžného prahování. Prosté prahování rozděluje pixely na dvě množiny. Pixely pod hodnotou prahu prohlásíme za pixely pozadí. Pixely dosahující alespoň hodnoty prahu jsou považovány za pixely proudnice. V ideálním případě by oblast pixelů proudnice měla být souvislá, spojitá. Velmi často se však uvnitř této oblasti objeví pixel, nedosahující hodnoty prahu. Typicky se tento jev objevuje na hranicích mlhy, kdy se hodnoty pixelů pohybují kolem prahové hodnoty.

Dalším problémem, který je typický pro prahování, je samotné stanovení prahové hodnoty. Pokud je práh nastaven příliš vysoko, splní jej až pixely z oblasti hustšího aditiva, nikoliv pixely na hranici oblasti. Naopak při nižším nastavení prahu hrozí, že jako mlha budou označeny i pixely mimo proud, kde může být přítomna neodvětraná ne příliš hustá mlha.

Oba popsané problémy lze vyřešit přidáním druhého prahu. Dvojice prahových hodnot P_1 , P_2 (kde $P_2 > P_1$) pak dělí pixely na tři skupiny:

- (a) Pixely, které jistě náleží mlze, dosahují alespoň hodnoty vyššího z prahů P_2 .
- (b) Pixely, které by mohly být mlhou, dosahují alespoň hodnoty nižšího z prahů P_1 .

(c) Pixely okolí, které nedosahují hodnot ani nižšího z prahů P_1 .

Po rozřídění pixelů do skupin nastává fáze, kdy jsou pixely prostřední skupiny (b) rozřazovány do skupin (a), (c). Procedura rozřazování vychází z myšlenky, že pokud by pixel mohl být pixelem mlhy (náleží skupině (b)), a přitom sousedí s pixelem mlhy, pak je pokládán za pixel mlhy. Rozřazovací proces se zaměřuje pouze na pixely skupiny (b). Spočívá v těchto krocích:

1. V první fázi je všem pixelům skupiny (a) dána značka „A“, všem pixelům skupiny (b) značka „B“ a všem pixelům skupiny (c) značka „C“. Tím je pro pixely zaveden druhý rozlišovací atribut.
2. Dále je postupně vyhodnocován každý pixel snímku.
 - 2.1. Pokud pixel skupiny (b) nesousedí s žádným pixelem skupiny (a), je předběžně označen jako pixel okolí (c), ale je mu ponechána značka „B“. Stále je tedy možné, že se stane pixelem mlhy, pokud se potvrdí, že nějaký jeho sousední pixel je pixelem mlhy.
 - 2.2. Pokud pixel skupiny (b) sousedí aspoň s jedním pixelem skupiny (a), je označen jako pixel mlhy (a) a dostává také značku „A“.
 - 2.3. Pokud byl pixel přerazen do skupiny (a), rekurzivně jsou stejným způsobem označeny jeho okolní pixely se značkou „B“. Ty mohou být ve skupině (b) či (c), ale značka jim ponechává možnost stát se pixely mlhy.
3. Po prohledání všech pixelů jsou tyto rozříděné pouze do skupin (a) a (c).

Na snímku 10 je znázorněn proces podmíněného prahování. Posloužil zde k odstranění neodvětrané řídké mlhy, která se nachází mimo proud. Snímek 10a představuje zdrojový obrázek. Snímek 10b znázorňuje výsledek prahování se dvěma prahy $P_1 = 85$ a $P_2 = 170$. Bílé oblasti jistě náleží mlze (skupina (a)), černé jistě náleží okolí (skupina (c)), šedé jsou pak pixely, které by mohly být mlhou (skupina (b)). Snímek 10c znázorňuje výsledek podmíněného prahování. Při hodnocení šedých oblastí skupiny (b) ze snímku 10b je patrné, že mlhou se staly ty z nich, které sousedily s bílou oblastí jisté mlhy. Naopak ty, které s jistou mlhou nesousedily, byly prohlášeny za okolí.



a) zdrojový snímek



b) snímek po prahování dvojicí prahů:
 $P_1 = 85$ a $P_2 = 170$



c) finální snímek po podmíněném prahování

Snímek 10 Podmíněné prahování

Zdroj: Odbor termomechaniky a techniky prostředí, VUT FSI [s.a.]

5.1.3 Vyhlašovaci algoritmus

Tento algoritmus bude sloužit k získání hranice detekovaného území a bude používán i v mnoha dalších metodách. Algoritmus vyžaduje, aby byl každý pixel snímku přiřazen do jedné ze dvou skupin, kdy skupina 0 („false“) představuje okolí a skupina 1 („true“) představuje mlhu. Právě prahování pixely do takových skupin rozděluje. Obecně může být oblast mlhy nespojitá, stejně jako oblast okolí.

Je možné, že nastanou situace, kdy je uprostřed oblasti mlhy detekován pixel okolí. Tento jev nastává například v případech, kdy aditivum není dostatečně husté, proudění je rychlé a turbulentní, v případě nevhodného zvolení prahu a podobně. Je velmi nepravděpodobné, aby pixel o těchto souřadnicích nebyl součástí proudu, a proto je nejjednodušším řešením jej do proudu také zahrnout, přestože jeho světelný odstín tomu neodpovídá.

Opačný jev, kdy je pixel mlhy detekován daleko od hlavního proudu, bývá způsoben přítomností mlhy, která prošla vyústkou již dříve a nyní již není součástí proudu, pouze v záběru zůstala jako neodvětraná. Z hlediska vyšetřování proudu je nežádoucí tyto oblasti uvažovat. Je zřejmé, že takové oblasti jsou co do počtu pixelů daleko menší než oblast hlavního proudu. Proto lze při konstrukci hranice uvažovat pouze tu největší detekovanou oblast pixelů mlhy. Pro konstrukci hranice je třeba stanovit a sledovat několik vlastností:

1. Nalezení počátečního pixelu hranice. Pro jednoduchost to může být pixel s nejmenší X souřadnicí. V případě většího počtu takových pixelů z nich vybereme ten s nejmenší Y souřadnicí.
2. Zvolení směru, kterým bude hranice konstruována – buď po směru či proti směru hodinových ručiček. V dalším bude uvažována konstrukce po směru hodinových ručiček.
3. Poslední pixel přidáný do hranice P_p . Na počátku je tímto bodem počáteční pixel hranice.
4. Sledovaným ukazatelem bude také vektor \mathbf{v}_p , v jehož směru byl přidán poslední pixel. Pro výchozí bod $[X_{min}, Y_{lib}]$ je $\mathbf{v}_p = [1, 0]$.

Přidání nového pixelu do hranice probíhá takto:

1. Jsou prohledávány sousední pixely pixelu P_p , počínaje bodem, který je v opačném směru vektoru \mathbf{v}_p . Pro výchozí stav je to vektor $-\mathbf{v}_p = [-1, 0]$.
2. Okolí bodu P_p je prohledáváno po směru hodinových ručiček. Výchozím směrem je $-\mathbf{v}_p$. Samotný pixel v tomto směru je ale přeskočen (je předchozím zařazeným bodem)
3. První nalezený pixel mlhy bude novým pixelem hranice, a jakožto poslední pixel do hranice přidáný se stává pixelem P_p . Zároveň určí nový vektor \mathbf{v}_p .

Pro příklad poslouží následující matice pixelů, kde 0 znamená pixel okolí a 1 znamená pixel mlhy:

Tabulka 3 Příklad pro demonstraci vyhlazovacího algoritmu

$Y \backslash X$	0	1	2	3	4
0	0	1	1	1	0
1	1	1	1	1	1
2	1	1	1	1	1
3	0	1	1	1	0

Konkrétní postup vypadá následovně:

1. Počátečním bodem hranice je zvolen $[0, 1]$. Je zároveň posledním bodem hranice P_p .
2. Poslední směr \mathbf{v}_p na počátku $[1, 0]$ a jemu opačný směr $-\mathbf{v}_p$ je $[-1, 0]$.
3. Pixely okolí bodu P_p jsou prohledávány ve směru hodinových ručiček. První pixel $[-1, 1]$ je vynechán. Další procházené pozice jsou $[-1, 0]$, $[0, 0]$, kde mlha není. Prvním nalezeným pixelem mlhy je pixel $[1, 0]$. Ten se stává novým pixelem P_p , vektor \mathbf{v}_p bude $[1, -1]$ a vektor $-\mathbf{v}_p$ pak $[-1, 1]$.
4. Okolí nového bodu $[1, 0]$ je prohledáváno. Pixel $[0, 1]$, který je ve směru $-\mathbf{v}_p = [-1, 1]$ je přeskočen. Prohledávány jsou pozice $[0, 0]$, $[0, -1]$, $[1, -1]$, $[2, -1]$, kde mlha není. Ta je nalezena až na pozici $[2, 0]$. Pixel $[2, 0]$ se stává novým pixelem P_p a určuje vektory $\mathbf{v}_p = [1, 0]$ a $-\mathbf{v}_p = [-1, 0]$.

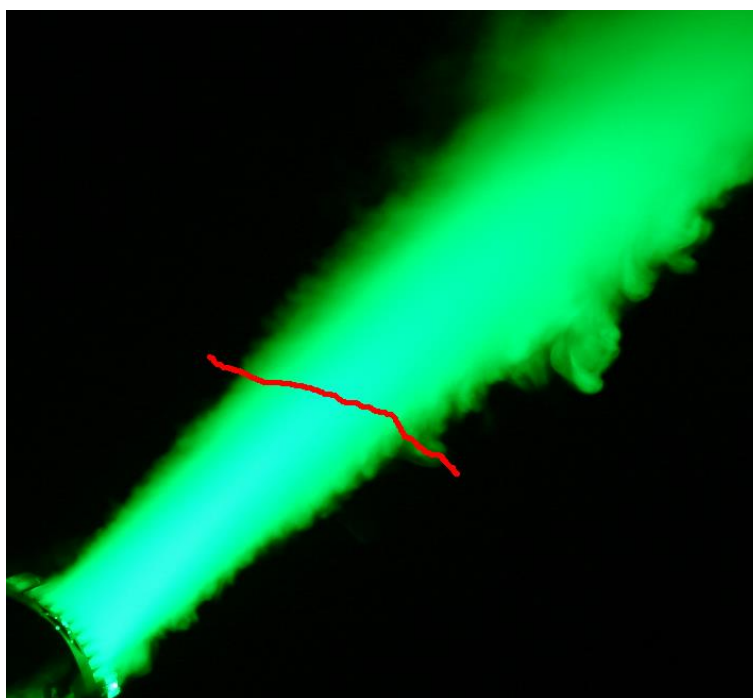
Celý algoritmus se opakuje až do návratu do pixelu, který byl vybrán jako první bod hranice.

Vyhlazovací algoritmus však nijak neupravuje hranici proudnice. Nezaobluje ji ani ji nedělá plynulejší. Tento úkol by si žádal další vývoj. Z kapitoly 9.1.3 však vyplývá, že takový algoritmus není pro účely této práce potřeba.

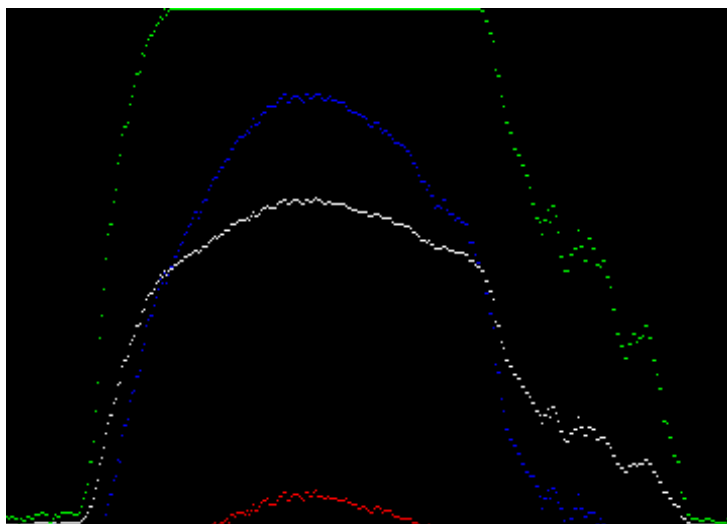
5.2 VYŠETŘOVÁNÍ PROUDNICE V ŘEZECH

Tento přístup prezentuje Caletka, Pech, Jedelský et al. (2015). Jedná se o metodu určování hranice proudu. Je vhodná především pro snímky prvního typu. Použitelné jsou i pro složitější snímky, avšak s homogenním pozadím. Dle Pavelka, Janotkové a Štětiny (2001) se řezem dvourozměrného snímku rozumí funkce $y = i(x)$, kdy definiční obor této funkce je dán pixely jakékoliv křivky, která celá leží uvnitř dotyčného snímku. Hodnoty funkce pak představují jednotlivé světelné intenzity pixelů, které tato křivka protíná. Popřípadě lze za řez považovat trojici funkcí s totožným definičním oborem, jejichž hodnoty představují intenzity jednotlivých barevných složek, tedy funkce $y_R = R(x)$; $y_G = G(x)$; $y_B = B(x)$.

Exemplární obrázek včetně křivky řezu lze vidět na snímku 11, vyhodnocení tohoto řezu pak na snímku 12.



Snímek 11 Exemplární snímek se znázorněnou křivkou řezu
Zdroj: Odbor termomechaniky a techniky prostředí, VUT FSI [s.a.]



Snímek 12 Vyhodnocení řezu ve snímku 11

Vysvětlení: *Bílý průběh* – jas; *Barevné průběhy* – jednotlivé intenzity barevných složek

V příkladu na snímcích 11 a 12 je použitý nazelenalý kouř. V takovém případě lze uvažovat o vyhodnocování například jen jeho zelené složky. Barevné složky mohou být užitečné také v případě, že pozadí má jinou než černou barvu, avšak barvu různou od barvy kouře. V případě snímku prvního typu lze považovat průběh jasu za dostatečně reprezentativní a vyhodnocení probíhá právě s hodnotami jasu.

Z průběhu hodnot na křivce lze vypořádat, že v bodě, kdy křivka prochází oblastí proudu, hodnoty vzrostou. Obecně platí, že mimo proud jsou hodnoty blízké nule (tmavé okolí). Na hranici proudu pak začnou narůstat v závislosti na zvyšující se hustotě látky. Ve středu proudu bývá hustota, a tedy i hodnota jasu maximální. Pro určení hranice proudu je klíčová právě oblast narůstajících hodnot. V ní očekáváme hraniční bod.

5.2.1 Vyhodnocení řezu

Pokud křivka řezu začíná a končí v pixelech pozadí a pokud vede oblastí proudu, je možné na ní detekovat dva hraniční body. Při detekci prvního bodu se postupuje od počátečního bodu křivky, při detekci druhého bodu se postupuje od jejího konce. Princip detekce je pro oba body totožný. Existuje více možností, jak lze detekovat bod, který budeme považovat za hraniční. Tyto možnosti popisují Richter a Šťastný (2012).

Detekce barevné intenzity

Pavelek, Janotková a Štětina (2001) uvádí tuto detekci s jediným parametrem, kterým je práh barevné intenzity. Pixely, jejichž intenzita dosáhne prahu, lze považovat za pixely v proudnici. Pixely, jejichž intenzita hranice nedosahuje, jsou považovány za pixely okolí. Postupuje se od krajního pixelu podél křivky, dokud barva nějakého pixelu nepřekročí práh intenzity. Tento bod je pak prohlášen za hraniční bod mlhy.

Hodnota intenzity bodu bývá stanovena s ohledem na jas celého snímku. Buď je odhadnuta, nebo stanovena konkrétní evaluační metodou (například průměrný jas snímku, x % z maximálního jasu ve snímku a podobně). Jiná možnost je provést evaluaci prahu pouze z hodnot pixelů konkrétní

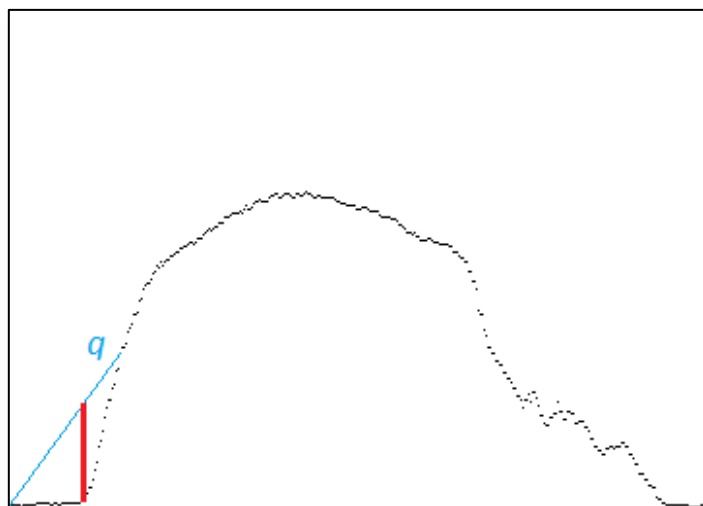
křivky řezu. Složitými výpočty nad pixely snímku by se však algoritmus připravil o svoji největší přednost, kterou je rychlost. V nejlepším případě, pokud konce křivky leží v oblasti proudu, pak dojde jen ke dvěma porovnáním. V nejhorším případě, pokud křivka o n pixelech obsahuje jediný pixel proudu, nebo leží zcela mimo proud, pak dojde k n porovnáním.

Klíčem ke správné detekci je vhodně určit práh intenzity. V případě, že bude práh příliš vysoký, nebude detekován pixel na hranici, ale až pixel v oblasti, kdy je proud dostatečně hustý. Pokud je naopak intenzita prahu jen nepatrně vyšší než intenzita okolí, je algoritmus často chybový, protože nesprávně detekuje hranici mimo proud, například díky přítomnosti parazitní neodvětrané vizualizační látky (řidké), která není součástí hustého proudu. Volba prahu je vždy kompromisní, a proto algoritmus detekce barevné intenzity nedosahuje kvalit sofistikovanějších algoritmů.

Detekce paty

Tato kapitola vychází z metod definovaných Pavelkem, Janotkovou a Štětinou (2001). Detekce paty probíhá ve dvou fázích. První fáze je totožná s algoritmem detekce barevné intenzity. Jejím účelem ale není hledat hraniční pixel, ale první pixel, který se s jistotou nachází v proudnici. Prah tedy bývá nastaven daleko výše než v předchozím případě. Může být určen konkrétní hodnotou, procentem z maximální intenzity v řezu, rozdílem oproti intenzitě okolí, procentem rozdílu oproti okolí apod. Způsob jeho evaluace však výsledek ovlivní jen minimálně. Druhá fáze postupu vychází z předpokladu, že krajní pixel křivky je pixelem pozadí. Pokud tato premisa neplatí, tedy krajní pixel náleží proudnici, je rovnou prohlášen za hraniční.

Pata mlhy je hledána právě dle prvního pixelu Q_0 křivky řezu a dle prvního detekovaného pixelu Q_i proudnice, konkrétně dle spojnice hodnot obou pixelů q . Úsečka q představuje pozvolný nárůst světelné intenzity. Skutečný nárůst intenzity je však odlišný. Pro pixely pozadí (ve snímku prvního typu) je jejich světelná intenzita přibližně konstantní. Jakmile se křivka dostane do oblasti mlhy, nastává skokový nárůst intenzity. Místo, kdy intenzita začne narůstat, je skutečnou hranicí mlhy. Pro dotýčný pixel platí, že rozdíl hodnoty jeho intenzity a hodnoty pro pozvolný nárůst intenzity (úsečka q) je maximální pro interval mezi body Q_0 a Q_i . Princip detekce paty je viditelný na snímku 13.



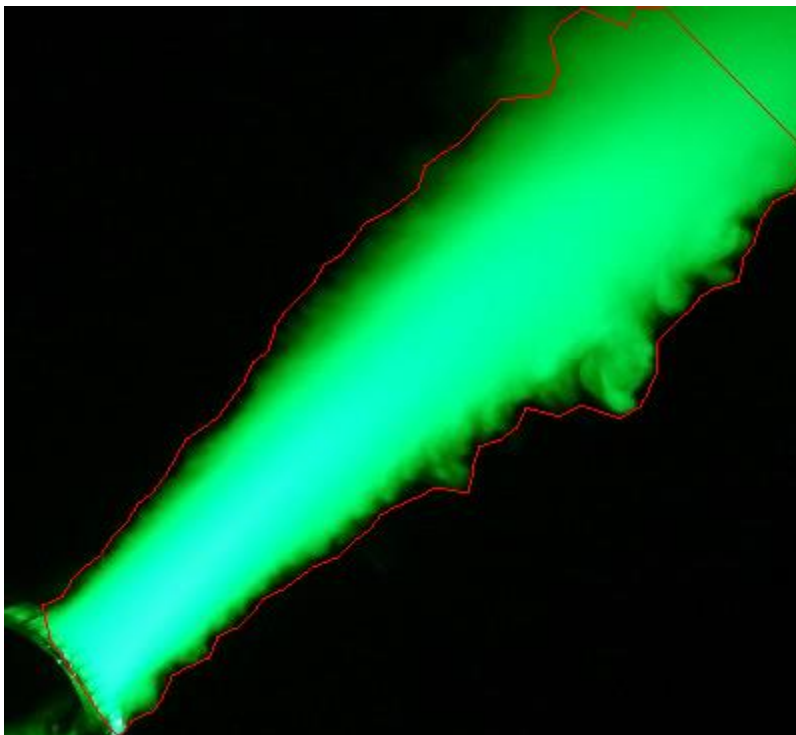
Snímek 13 Detekce paty v průběhu jasu

5.2.2 Sestrojení hranice proudu

Jeden řez tedy slouží k určení dvou hraničních bodů, přičemž jeden náleží horní hranici proudu, druhý náleží spodní hranici proudu. Jak uvádí Pavelek, Janotková a Štětina (2001), pro komplexní přehled o tvaru proudnice, je třeba provést vyhodnocení v po sobě jdoucích přímých řezech napříč snímkem, které protínají proud, pokud možno, co nejvíce kolmo na směr proudění. Přímký řezu budou vzájemně rovnoběžné a jejich vzdálenost definuje zvolený parametr d .

Z každého řezu y_i je získána dvojice hraničních bodů H_{Ai} , H_{Bi} . Bod H_{Ai} bude přiřazen k hranici B_A , a bude spojen úsečkou s jejím předchozím H_{Ai-1} . Bod H_{Bi} bude obdobně zařazen do hranice B_B . Pokud žádný bod hranice detekován není, celý řez je přeskočen. Po zpracování stanovených řezů snímku budou kompletní křivky hranic B_A a B_B . Posledním krokem je spojit úsečkou první body H_{A0} a H_{B0} a poslední body H_{An-1} a H_{Bn-1} . Tím se obě hranice sjednotí v jednu a proudnice je tím ohraničena.

Výsledek detekce hranice ve snímku 11 je na snímku 14. Řezy byly vyhodnocovány po vzdálenosti 10 pixelů ve směru $\pi/4$, a to metodou detekce paty s parametrem 100.



Snímek 14 Hranice sestrojená vyhodnocením po sobě jdoucích řezů

Výhodou této metody je možnost detekovat i přítomnost velmi nevýrazného aditiva. Nevýhodou je její použitelnost pouze na snímky prvního typu. Vizualizovaná látka musí být od okolí dobře rozeznatelná a proud se musí šířit neměnným hlavním směrem.

5.3 ROZPOZNÁVÁNÍ MLHY NEURONOVÝMI SÍTĚMI

Tímto způsobem detekce se zabývá Richter, Šťastný a Jedelský (2013). Metoda vychází z poznatků kapitoly 5.1. V té je použito jednoho či více prahů k určení, zda se v konkrétním pixelu nachází mlha, či nikoliv. Takovou informaci může být někdy obtížné usuzovat pouze na základě barevného odstínu pixelu, zejména u snímků s heterogenním pozadím. Proto detekce mlhy prahováním funguje spolehlivě pouze pro snímky prvního typu. Snímky komplikovanější už mohou obsahovat světlé pixely, které však nenáleží mlze a jejich chybné prohlášení za mlhu může velmi negativně ovlivnit kvalitu výsledku.

Pokud však vyhodnocení provádí člověk, dokáže mlhu i tak rozeznat. Neděje se tak podle jasu jediného pixelu. Na určité malé oblasti už však člověk dokáže rozhodnout, zda tato oblast obsahuje mlhu či nikoliv. Při porovnání oblasti mlhy a oblasti jiného světlého prostředí je velmi obtížné exaktně popsat, jak je od sebe rozlišit. Přesto je tohoto rozpoznávání člověk schopen. Vyhodnocení člověkem tedy může inspirovat k odlišnému přístupu k detekci mlhy, a to pomocí umělých neuronových sítí.

5.3.1 Princip vyhodnocení neuronovou sítí

Jak již bylo řečeno, člověk z malého okolí pixelů dokáže rozhodnout (nikoliv vždy správně), zdali toto okolí náleží mlze či nikoliv. Stejná schopnost bude požadována po umělé neuronové síti, jak to popisuje Richter, Šťastný a Jedelský (2013).

Dle Richtera, Šťastného a Jedelského (2013) je vhodným typem neuronové sítě pro takový účel vícevrstvá dopředná neuronová síť *MLP*. Jejími vstupy budou pixely malého okolí vyšetřovaného pixelu, ze kterého lze co do velikosti rozhodovat o detekci mlhy. Vhodná velikost okolí vyšetřovaného bodu je 5×5 nebo 7×7 pixelů (prostřední pixel je vyšetřovaný). Nabízí se tedy 25 (respektive 49) vstupních neuronů sítě, přičemž na každý bude přivedena hodnota jasu příslušného pixelu. Protože však předpokládáme použití tohoto přístupu na snímky s heterogenním pozadím, hraje roli nejenom jas pixelů, ale i jejich barevný odstín. Pro snímky druhého a zejména třetího typu je vhodné předat neuronové síti *RGB* intenzity všech 25 (49) pixelů. Potřebných vstupů tak musí být trojnásobek, tedy 75 (147).

Výstupní vrstva bude obsahovat právě jeden neuron, který bude odpovídat na otázku, zda je prostřední pixel oblasti pixelem mlhy či nikoliv. Výstupem tohoto neuronu bude hodnota v rozmezí $0 - 1$. Čím vyšší hodnota bude, tím více je středový pixel sítě pokládán za pixel mlhy. Hodnotu tak lze chápat jako pravděpodobnost sledovaného jevu.

Mezi vstupní a výstupní vrstvou může být jedna či více vrstev skrytých. Konkrétní navržení topologie neuronové sítě je úkolem pro testování. Pro síť, kde je na vstupy předáván jas pixelů, postačí jediná skrytá vrstva s mírně navýšeným počtem neuronů oproti vrstvě vstupní, zhruba o 20 %. Vhodnou sítí tak může být 25-30-1 respektive 49-60-1. Pro řešení s *RGB* vstupy je však jistější použít více skrytých vrstev, neboť se předpokládá zhodnocování jednoho kroku navíc, kterým je zhodnocení odstínu pixelu. Je totiž zřejmé, že odstín mlhy je blízký odstínu šedi, tedy intenzity jednotlivých barevných složek bývají podobné. Naopak barevné složky pixelů, kde se mlha nenachází, se mohou v hodnotě intenzity velmi lišit. Proto se jako vhodné jeví přidat neuronové síti další skrytou vrstvu. Pro *RGB* vstupy jsou použitelné sítě například 75-90-30-1 nebo 147-180-60-1.

Jakmile je neuronová síť navržena, je třeba ji naučit mlhu rozpoznávat, a to tak, že jí budou předkládány vzory oblasti mlhy, kdy je požadovanou hodnotou výstupu 1, a vzory oblasti okolí s požadovanou hodnotou výstupu 0. Je vhodné provádět proces učení na snímku, který budeme vyhodnocovat nebo na podobném ze stejného prostředí (například jiný snímek sekvence proudění, pořízeného statickou kamerou či fotoaparátem, který snímá stále stejný prostor).

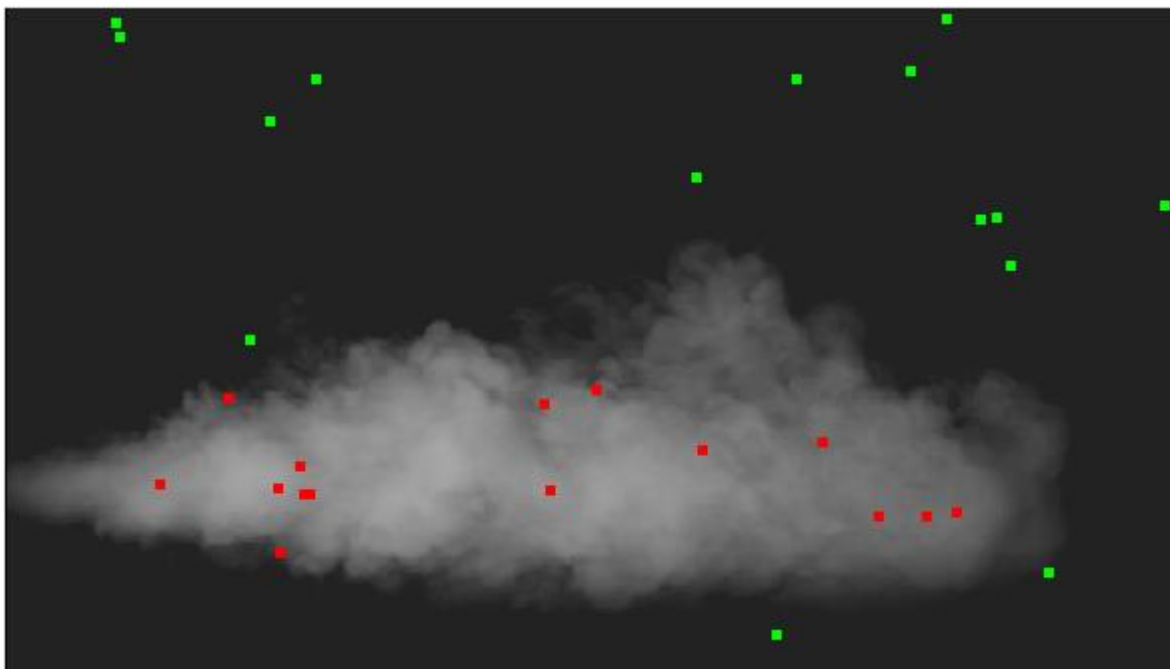
5.3.2 Základní zpracování snímku s využitím neuronové sítě

V kapitole 5.3.1 byla navržena neuronová síť, která hodnotí, na kolik % pokládá vyšetřovaný pixel za pixel mlhy. V základním vyhodnocení (Richter, Šťastný, Jedelský, 2013) bude za pixel mlhy považován každý pixel, o kterém neuronová síť prohlásí, že je pixelem mlhy s pravděpodobností alespoň 50 %.

Pokud bude každý pixel snímku podroben rozhodovacímu procesu s využitím neuronové sítě, výsledkem procedury bude snímek pravdivých (označovány bílou) a nepravdivých (označovány černou) pixelů. Tedy výsledek stejného typu jako výsledek po operaci prahování. Oblast mlhy je pak vhodné ohraničit postupem vyhlazovacího algoritmu popsáním v kapitole 5.1.3.

Princip učení demonstruje snímek 15. Protože se jedná o snímek prvního typu, postačí neuronová síť 25-30-1 zpracovávající jas pixelů oblasti 5×5 . Snímek 15 zachycuje výběr vzorů, které se bude neuronová síť učit. Červeně jsou znázorněny vzory s požadovanou hodnotou 1, zeleně vzory s požadovanou hodnotou 0. Vzory určuje člověk buď přímo, nebo je síti předáno již hotové vyhodnocení člověkem a v něm jsou body mlhy a okolí vybrány náhodně.

Jakmile je síť naučena, každému pixelu snímku $[X, Y]$ je vytvořena oblast $[X-2...X+2, Y-2...Y+2]$ a jednotlivé pixely jsou předány příslušným vstupům sítě. Pokud je na výstupním neuronu obdržena hodnota větší nebo rovna 0.5, bod $[X, Y]$ je prohlášen za bod mlhy, v opačném případě za bod okolí. Na snímku 16 je znázorněn výsledek celého vyhodnocení. Bílé pixely jsou ty, které neuronová síť označila hodnotou alespoň 0.5. Pro černé pixely byla obdržena hodnota nižší.



Snímek 15 Předání vzorů neuronové síti



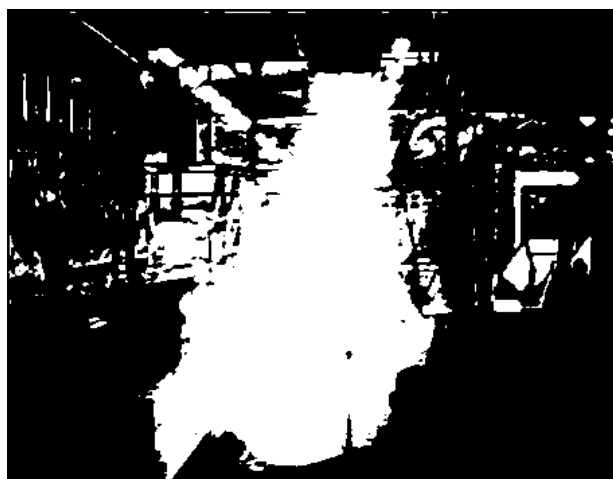
Snímek 16 Vyhodnocení snímku prvního typu neuronovou sítí

V případě snímků prvního typu nelze říci, že je vyhodnocení neuronovou sítí kvalitnější v porovnání s prostým prahováním. U snímků komplikovanějších už však prahování nedosahuje potřebné kvality, zato neuronová síť je zde úspěšnější.

Snímek 17 znázorňuje proces detekce mlhy neuronovou sítí ve snímku třetího typu s heterogenním pozadím. Použita byla síť 75-90-30-1 zpracovávající *RGB* vstupy. Na snímku 17a je zdrojový snímek, na němž se síť učí. Na snímku 17b je pak vyhodnocení takového snímku touto sítí. Pro ilustraci je pak na snímku 17c výsledek vyhodnocení sítí 25-30-1, která zpracovává pouze jas pixelů. Je patrné, jak velkou výhodou je možnost uvažování jednotlivých barevných složek.



a) zdrojový snímek



b) vyhodnocení dle *RGB* sítí 75-90-30-1



c) vyhodnocení dle jasu sítí 25-30-1

Snímek 17 Vyhodnocení snímku třetího typu neuronovou sítí

Výsledek po vyhlazovacím algoritmu je znázorněn na snímku 18.



Snímek 18 Detekovaná mlha po aplikaci vyhlazovacího algoritmu

Takto naučená síť je pak vhodná nejen k vyhodnocování snímků z téže sekvence, ale dokonce podobných snímků ze stejného prostředí, podobné světlosti a s podobným heterogenním pozadím. To znázorňují příklady na snímku 19.



a) zdrojový snímek I



b) výsledek detekce na snímku I



c) zdrojový snímek II



d) výsledek detekce na snímku II

Snímek 19 Detekce mlhy sítí, naučenou na snímku 17a, v podobných snímcích
Zdroj: Odbor termomechaniky a techniky prostředí, VUT FSI [s.a.]

Z výsledků je však patrné, že i přes dodatečný vyhlazovací algoritmus byly do oblasti mlhy nesprávně zahrnuty i další oblasti s podobnou barvou, zejména světlá místa jako lesklá podlaha, lesklá roura či osvětlení. Na snímku 20 je porovnání barev obou okolí, přičemž by bylo i pro člověka obtížné mlhu v tomto případě odlišit pouze na základě takových dat. Další kapitoly budou věnovány možnostem, jak parazitní oblasti z výsledné oblasti vyloučit.



a) skutečná mlha



b) lesklá podlaha

Snímek 20 Porovnání oblastí podobné barvy
Zdroj: Richter, Šťastný, Jedelský, 2013

5.3.3 Podmíněné prahování spojitě detekce

Pokud člověk provádí detekci mlhy, nesoustředí se na jednotlivé body odděleně. Nejprve najde místa, v nichž si je mlhou zcela jistý. Následně hledá, kam až mlha sahá. Pouze dle barevné struktury by si člověk oblastmi na hranici mlhy jistý nebyl, méně hustou mlhou již prosvítá barva pozadí za ní. Protože však ví, že rozšiřuje oblast mlhy, přisoudí texturu také mlze. Takový postup lze napodobit využitím spojitě detekce. V kapitole 5.3.1 bylo uvedeno, že výstupem neuronů je hodnota v rozmezí 0-1. Pro základní zpracování snímku s využitím neuronové sítě by dostačovala neuronová síť s výstupem 0 nebo 1. Spojité rozmezí výstupu však bude využito právě při spojitě detekci.

Prvním krokem spojitě detekce je opět vyhodnocení okolí každého pixelu snímku neuronovou sítí. Každý pixel $[X, Y]$ tak obdrží hodnotu N_{XY} v rozmezí 0-1, která však nyní nebude prahována hodnotou 0.5. Namísto toho je využit postup podmíněného prahování, zmíněný v kapitole 5.1.2. Pro

tento postup bude existovat dvojice prahů P_1 , P_2 , kde $P_2 > P_1$. Pixely jsou podle dotyčných výstupů neuronové sítě rozděleny dle prahů na 3 skupiny:

- (a) $N_{XY} \geq P_2$ – Pixely, které dle výstupu neuronové sítě s jistotou náleží mlze.
- (b) $P_2 > N_{XY} \geq P_1$ – Pixely, které by dle výstupu mohly také náležet mlze.
- (c) $P_1 > N_{XY}$ – Pixely, které jistě náleží okolí

Algoritmus zmíněný v kapitole 5.1.2 pak simuluje rozpoznávání mlhy člověkem, kdy je jako první označena oblast husté mlhy (a), která je následně rozšiřována o sousední oblast (b), jejíž barva již není tak průkazná. Právě proto, že by dle barvy mohla být mlhou a že s mlhou sousedí, je označena za mlhu také.

Poznámka: Na výsledek podmíněného prahování byl použitý tentýž vyhlazovací algoritmus jako v případě základního zpracování.

Na snímku 21 je porovnání vyhodnocení základního zpracování neuronovou sítí (snímek 21b) a spojitě detekce neuronovou sítí (snímek 21c). Prahování základního zpracování neuronové sítě bylo prováděno hodnotou 0.5. Prahy po spojitě detekci byly $P_1 = 0.3$ a $P_2 = 0.7$. Spojitá detekce mnohem méně chybí v tom, že oblast označí jako mlhu. Rovněž lépe detekuje hranice, jelikož do mlhy zahrnuje i oblasti, jejichž barva není tak průkazná. Díky vyššímu prahu P_2 některé oblasti vůbec neuvažuje, protože tento práh nepřekonají. Jiné oblasti naopak zvýhodňuje díky nízkému prahu P_1 , právě proto, že tyto oblasti s mlhou sousedí. Při porovnání výsledků obou detekcí lze pozorovat, že především díky možnosti uvažovat i méně hustou mlhu znamená spojitá detekce se dvěma prahy vylepšení.

Podmíněná detekce však stále neřeší problém, že některé oblasti okolí byly s jistotou označeny jako mlha díky barvě velmi podobné mlze (na snímku 21 je to především roura a kousek šedé podlahy). V některých případech může být také rozšíření nežádoucí. Pokud mlha proudí před oblastí pozadí, která by však dle barvy mohla být mlhou, detekovaná oblast se pak rozšíří i na ni.

Poznámka: Snímku 21a již byla mlha detekována (viz snímek 19). Avšak síť v tom případě byla trénována na jiném snímku.



a) zdrojový snímek



b) výsledek základní detekce neuronovou sítí



c) výsledek spojitě detekce neuronovou sítí

Snímek 21 Porovnání základní a spojitě detekce mlhy neuronovou sítí

5.3.4 Zvětšení neuronové sítě

Předchozí přístupy často vedly k tomu, že za mlhu byla označena i oblasti pozadí s barvou mlze velmi podobnou. Používané neuronové sítě nedokážou tyto drobné rozdíly spolehlivě odhalit. Jistou možností by bylo zvýšit počet neuronů v neuronové síti tak, aby dokázala obsáhnout větší počet vzorů, které si však budou velmi podobné.

K pokusu byla použita místo sítě s neurony 75-90-30-1 síť s dvojnásobným počtem neuronů ve skrytých vrstvách, tedy 75-180-60-1. Byly jí předávány vzory mlhy i okolí, které však často měly velmi podobnou barvu. Snímek 22 porovnává výsledek původní sítě 75-90-30-1 (snímek 22b) a rozšířené 75-180-60-1 (snímek 22c). K určitému zlepšení došlo, síť například detekuje také ne příliš hustou mlhu napravo od vyústky. Oblast lesklé podlahy již nebyla označena celá, ale pouze její část. K úplnému potlačení chybné detekce lesklé podlahy však nedošlo. Otázkou také zůstávají lesklá místa za mlhou, konkrétně zářivky na stropě haly. Síť byla naučena, že příliš lesklá místa mlhou nejsou. V tomto případě by člověk oblast zářivek z mlhy nevytloučil, protože se mlha nachází všude okolo.



a) zdrojový snímek



b) 75-90-30-1

c) 75-180-30-1

Snímek 22 Porovnání výsledků různých neuronových sítí

5.3.5 Metoda dvou expertů

Jak vyplývá z kapitoly 5.3.4, je obtížné naučit neuronovou síť rozpoznávat jednoznačné barevné okolí a mlhu a zároveň mlhu a velmi podobnou barvu (lesk a podobně). Ani pro dvojnásobný počet neuronů skrytých vrstev se problém úplně nevyřešil.

Namísto dalšího zvětšování počtu neuronů se nabízí možnost použití dvou různých neuronových sítí. Principiálně by obě fungovaly stejně, avšak každá by byla naučená na jiný typ rozpoznávání:

- 1) **Obecný expert** – Neuronová síť bude expertem na rozlišování oblastí okolí a oblastí, které by co do barevnosti mohly být mlhou.
- 2) **Specifický expert** – Neuronová síť se soustředí pouze na oblasti, které by mohly být mlhou. Ty třídí na mlhu a mlze podobné okolí.

Obě sítě mohou mít stejný počet neuronů i strukturu. Obě by měly zpracovávat *RGB* vstupy (důležité zejména pro specifického experta). Jako obecného experta lze použít neuronovou síť, které se věnuje kapitola 5.3.2 a která je využívána i v kapitole 5.3.3. Její neschopnost rozlišit například lesklá místa od mlhy bude kompenzovat specifický expert.

Postup naučení obecného experta je tedy známý. Specifický expert je však učen pouze na oblasti, kterou obecný expert označil jako „možnou mlhu“. Snímek 23 zachycuje vyhodnocení obecným expertem dle postupu kapitoly 5.3.2. Snímek 23a je zdrojovým snímkem, snímek 23b zachycuje finální oblast, kterou síť označila jako možnou mlhu (v původních barvách).



a) zdrojový snímek



b) barevně oříznutá oblast možné mlhy

Snímek 23 Vyhodnocení obecným expertem

Specifický expert se bude učit pouze v oblasti, kterou obecný expert označil za možnou mlhu, tedy v nečerné oblasti. S hodnotou 1 jí budou předány oblasti kolem bodů mlhy, s hodnotou 0 pak oblasti lesklé podlahy. O místech mimo oblast možné mlhy však specifický expert vůbec nebude umět rozhodnout, protože se na ně nespecializuje. Principem metody je, že specifický expert pracuje právě jen na oblasti, která by mohla být mlhou dle obecného experta. Z této oblasti vyseparuje ta místa pozadí, která první síť chybně pokládala za mlhu. Výsledek detekce mlhy dvojicí expertů je na snímku 24a. Získaná hranice je vykreslena do původního obrázku na snímku 24b.



a) výsledek detekce



b) znázorněná hranice v původním obrázku (tyrkysově)

Snímek 24 Vyhodnocení specifickým expertem

Je vhodné poznamenat, že stejný výsledek by byl získán, pokud by celý snímek vyhodnotil obecný i specifický expert. Jako mlha by byla označena pouze místa, na kterých se oba experti shodnou, tedy oba je označí za mlhu. Pomocí dvojice expertů lze tedy potlačit neschopnost rozlišit mlhu a mlze barevně velmi podobné oblasti.

5.3.6 Podmíněná detekce dvojicí expertů

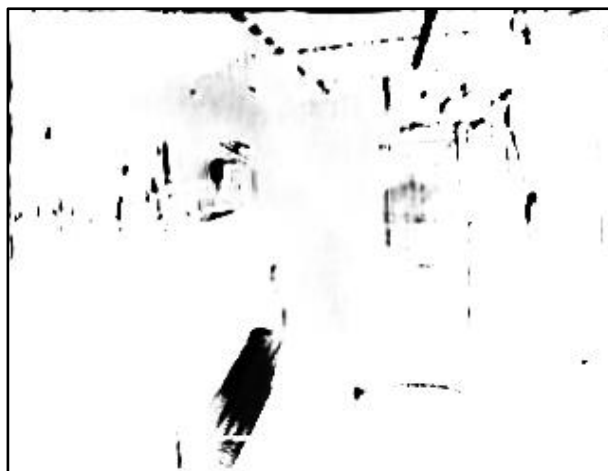
Úskalí předchozí metody mohou znamenat místa, kde mlha obklopuje mlze podobné okolí. Například v místech, kde je mlha prosvícena zářivkami. Specifický expert má za úkol zářivky od mlhy odlišit. Protože jsou však zářivky mlhou obklopeny, mlha bude pravděpodobně i na jejich místě, přestože světlo zářivek mlhu prosvítí a ta tak není patrná. Je však zřejmé, že tento problém nelze uspokojivě vyřešit pouze pomocí hodnocení barev. Člověk v takovém případě posuzuje okolí, v němž se zářivý pixel nachází, a pokud se v okolí mlha nachází a světelná intenzita oblasti mlhu nevyklučuje, pak člověk oblast za mlhu považuje. Touto myšlenkou byla inspirována již metoda v kapitole 5.3.3. Její princip lze zkombinovat právě s metodou dvojice expertů.

Oba experti tedy budou detekovat spojitě. Výsledek spojitě detekce oběma experty ve snímku 23a je na snímku 25. Protože však byl specifický expert trénován pouze na oblasti označené obecným expertem za mlhu, okolí této oblasti vůbec nezná a mohl se zde rozhodnout jakkoliv. V tomto konkrétním případě lze říci, že lesklá místa označuje jako hodnotou 0 (přibližně), a proto například pozadí továrny a okolní objekty označil hodnotami blízkými 1.

Poznámka: Protože je účelem expertů řešení týchž problémů jako v kapitole 5.3.5, jsou také trénováni na stejných typech vzorů. Lze použít dokonce tytéž neuronové sítě.



a) detekce obecným expertem



b) detekce specifickým expertem

Snímek 25 Spojitá detekce

Existují dvě možnosti, jak aplikovat podmíněné prahování:

Podmíněné prahování jednotlivých výsledků

Spojitě detekované snímky jsou podrobeny podmíněnému prahování, čímž je v obou z nich definováno, co je a co není považováno za mlhu. V obou případech jsou voleny velmi přísné prahové intenzity $I_1 = 200$ ($P_1 = 78\%$), $I_2 = 250$ ($P_2 = 98\%$). Snímek, na němž vyhodnocoval obecný expert, je možné zpracovat vyhlazovacím algoritmem. Naprahaný výsledek specifického experta znázorňuje oblasti, které by mohly být mylně označovány za mlhu. Nezobrazuje konkrétní proudnici, a proto u něj vyhlazovací algoritmus nelze aplikovat. Snímek 26 zachycuje oba mezivýsledky před tím, než budou sloučeny.



a) snímek po vyhodnocení obecným expertem, po podmíněném prahování a po vyhlazovacím algoritmu



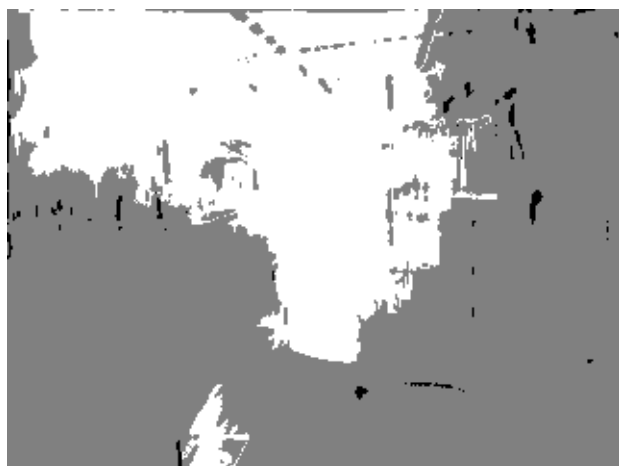
b) snímek po vyhodnocení specifickým expertem a po podmíněném prahování

Snímek 26 Vyhodnocení dvojicí expertů před sloučením

Sloučení snímků proběhne jako v kapitole 5.3.5, tedy pokud bude konkrétní pixel v obou snímcích označen jako pixel mlhy, pak je skutečným pixelem mlhy. Snímek 27a zachycuje sloučený snímek před aplikací zmíněné podmínky, kdy:

- Černé oblasti nebyly ani v jednom zdrojovém snímku označeny jako mlha.
- Šedé oblasti s intenzitou 127 byly označeny jako mlha právě v jednom snímku.
- Bílé oblasti byly jako mlha označeny v obou zdrojových snímcích.

Konečný výsledek po opětovném vyhlazovacím algoritmu je na snímku 27b.



a) bez úprav



b) po vyhlazovacím algoritmu nad body, které oba experti označili za mlhu

Snímek 27 Výsledek sloučení jednotlivých prahovaných snímků

Použití podmíněného prahování jednotlivých výsledků má však jedno úskalí, a to v použití podmíněného prahování na snímek zpracovaný specifickým expertem. Účelem obecného experta je označit místa, která by mohla být mlhou. Účel specifického experta je však spíše opačný, a to

vyseparovat z výsledků obecného experta oblasti, které mlhou nejsou. Je tak otázkou, jestli má podmíněné prahování v případě tohoto snímku případně smysl.

Podmíněné prahování sloučeného výsledku

Jak název napovídá, prvním krokem postupu je sloučit výsledky spojitých detekcí obou expertů, tj. snímky 25a a 25b.

Jak bylo řečeno, jas pixelu je úměrný hodnotě 0-1, která je na výstupu neuronové sítě: jas 0 (černá) odpovídá návratové hodnotě 0 a jas 255 (bílá) odpovídá návratové hodnotě 1. Pokud má být při slučování snímků oběma výsledkům přikládána stejná váha, pro oba bude platit koeficient 0.5. Pro každý pixel finálního snímku $P_F[X, Y]$ je stanovena světelná intenzita jako

$$I_F[X, Y] = 0.5 \cdot I_O[X, Y] + 0.5 \cdot I_S[X, Y] \quad (5.1)$$

Kde I_F je intenzita pixelu ve finálním snímku

I_O je intenzita příslušného pixelu ve snímku vyhodnocení obecným expertem

I_S je intenzita příslušného pixelu ve snímku vyhodnocení specifickým expertem

Výsledek sloučení snímků je na snímku 28.



Snímek 28 Snímek sloučený z dílčích vyhodnocení oběma experty

V předchozí variantě byla jako mlha označena pouze místa, která byla v obou dílčích výsledcích označena jako mlha. I nyní tedy jsou zvolena velmi přísná nastavení prahových intenzit $I_1 = 200$ ($P_1 = 78 \%$), $I_2 = 250$ ($P_2 = 98 \%$). Mlha může být označena, pouze pokud si jsou obě sítě nějakou její oblastí takřka úplně jisté. Rovněž oblasti, do kterých pak může být rozšířena, musí splňovat přísnou pravděpodobnostní podmínku. Výsledek celé metody (po aplikaci vyhlazovacího algoritmu) je patrný na snímku 29.



Snímek 29 Výsledek podmíněného prahování sloučeného výsledku

5.3.7 Další využití detekce neuronovými sítěmi

Popsaný princip detekce neuronovou sítí je v rámci této práce využíván k detekci mlhy. Je však možné jej použít i k řadě dalších účelů. V kapitole 4.2 byla neuronová síť použita coby hranový detektor. Lze ji též aplikovat na detekci nejrůznějších textur v obraze. Na snímku 30 lze vidět detekci koberce na chodbě. Výsledek detekce je znázorněn červeně.

Poznámka: Protože byl na detekci aplikován vyhlazovací algoritmus, detekovaná oblast je spojitá. Proto není zvýrazněna trojúhelníková oblast koberce pod schodištěm.



a) vyhodnocovaný snímek



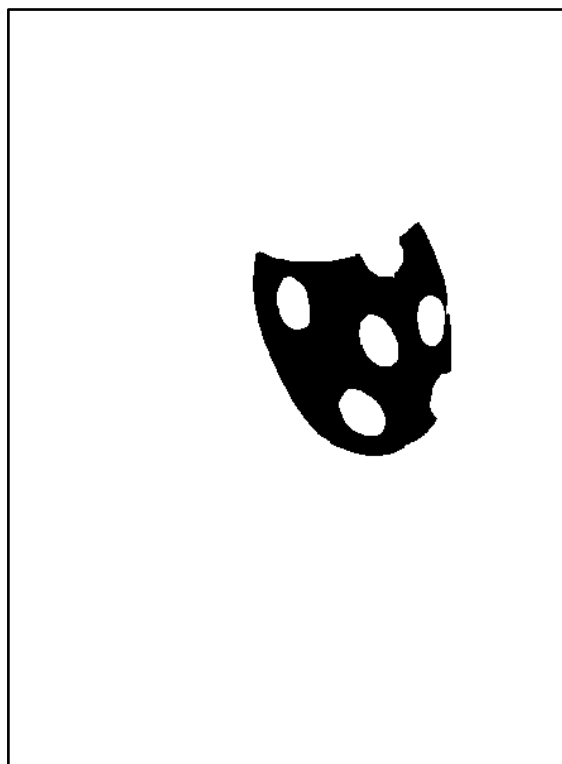
b) výsledek vyhodnocení (koberec je označen červeně)

Snímek 30 Detekce koberce

Velmi účinné je využít neuronovou síť na detekci přibližné barvy, například při přebarvování ilustrací. Na snímku 31a je obrázek Berušky, v němž je třeba detekovat oblast červených krovek. Nelze však použít detekci konkrétního barevného odstínu, protože červená barva není homogenní. Pokud však bude neuronová síť naučena, co je a co není červená, lze s ní poté oblast krovek detekovat. Na snímku 31b je detekovaná oblast.



a) vyhodnocovaný snímek



b) výsledek vyhodnocení

Snímek 31 Detekce červených krovek Berušky
Zdroj: rajce.net, 2014

6 URČENÍ TVARU PROUDU VIZUALIZOVANÉHO HELIOVÝMI BUBLINAMI

Tvar proudnice je patrný i ze snímku vizualizace heliovými bublinami. Postup získání proudnice je však zcela odlišný od snímků proudů mlhy, protože heliové bubliny tvoří souvislá vlákna, nýbrž částice. Jedná se o velmi malé objekty, které lze teoreticky pokládat za body. Proudnice však není pouze v místě bublin samotných, ale také v prostoru mezi nimi. Určení tvaru proudy tak bude sestávat ze dvou kroků, a to vyhledání bublin a odvození tvaru dle jejich poloh. Základní myšlenky tohoto postupu uvádí Šťastný a Richter (2016).

6.1 DETEKCE HELIOVÝCH BUBLIN NÁLEŽÍCÍCH PROUDU

Účelem tohoto kroku bude správné označení bublin v proudě a jejich zjednodušení na body. Díky tomu, že heliové bubliny budou považovány za body, bude možné v operacích s nimi použít vybrané metody teorie grafů. Také hlavní vyhodnocovací algoritmus bude pracovat s řádově menším množstvím vstupních dat, což podstatně sníží časovou náročnost celého procesu.

Tato kapitola se soustředí především na snímky, v nichž jsou heliové bubliny velmi dobře patrné, ideálně na snímky prvního typu.

6.1.1 Určení pixelů náležících heliovým bublinám

Ve snímcích prvního typu jsou bubliny velmi dobře odlišitelné od heterogenního pozadí. Díky tomu lze snadno zautomatizovat proces určení, které pixely náleží bublinám a které okolí. Nejprve je snímek statisticky zhodnocen, čímž definuje parametry následné detekce. Ve snímku je určen rozdíl největší a nejmenší hodnoty pro každou barevnou složku:

- D_R – Rozdíl největší a nejmenší hodnoty R složky ve snímku
- D_G – Rozdíl největší a nejmenší hodnoty G složky ve snímku
- D_B – Rozdíl největší a nejmenší hodnoty B složky ve snímku

Předpokladem metody je, že bubliny jsou od heterogenního pozadí dobře rozlišitelné. Proto se budou extrémy alespoň jedné z barevných složek značně lišit. Ze všech rozdílů je vybrán rozdíl největší. Podle dotyčné barevné složky budou pixely rozdělovány na pixely okolí a pixely bublin. V následujícím příkladu bude uvažováno, že největší rozdíl zaznamenala barevná složka B .

V dalším kroku je stanovena průměrná hodnota modré barevné složky B_{avg} . Počet pixelů snímku je známý a sumu hodnot B všech pixelů lze počítat už při předchozím kroku hledání extrémů. Protože pixelů heliových bublin je mnohem méně než pixelů pozadí, B složka barvy pozadí je ta, k níž je průměr B_{avg} blíže. Schematicky to znázorňuje příklad na snímku 32, kde rozdíl $B_{max} - B_{avg} < B_{avg} - B_{min}$. Hodnota B_{max} tedy v tomto případě náleží pozadí a hodnota B_{min} náleží bublině.

Zbývá určit prahovou hodnotu B_t , pomocí které budou pixely rozdělovány na pixely bublin a pixely okolí. Dalším předpokladem bude, že hodnota B_{min} byla zaznamenána na bublině a hodnota B_{max} na okolí. Pro prahovou hodnotu se nabízí střed mezi B_{min} a B_{max} :

$$B_t = \frac{B_{min} + B_{max}}{2} \quad (6.1)$$

Snímek však může obsahovat také množství šumu i nevýrazných bublin, jako na snímku 33a. V takovém případě je vhodné práh zvolit blíže k B_{min} , aby všechny nežádoucí elementy nebyly započítány mezi bubliny. Zpřísnění prahu je docíleno například vzorcí:

$$B_t = \frac{B_{avg} + B_{max}}{2} \quad (6.2)$$

či

$$B_t = B_{min} + (B_{max} - B_{avg}) \quad (6.3)$$

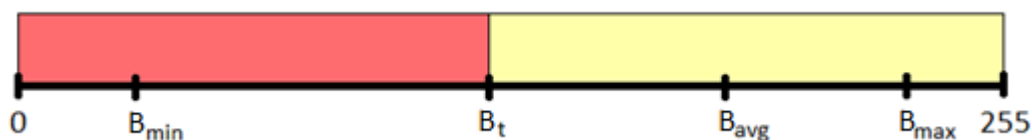
ale pro další použití bude použitý obecný předpis

$$B_t = B_{min} + k \cdot (B_{max} - B_{min}) \quad (6.4)$$

kde $k = \langle 0, 1 \rangle$ je voleno experimentálně, pro zpřísnění prahu $k < 0.5$.

Pokud $B_{avg} > B_t$, pak každý pixel P_{XY} , jehož modrá složka $B_{XY} < B_t$, náleží bublině.

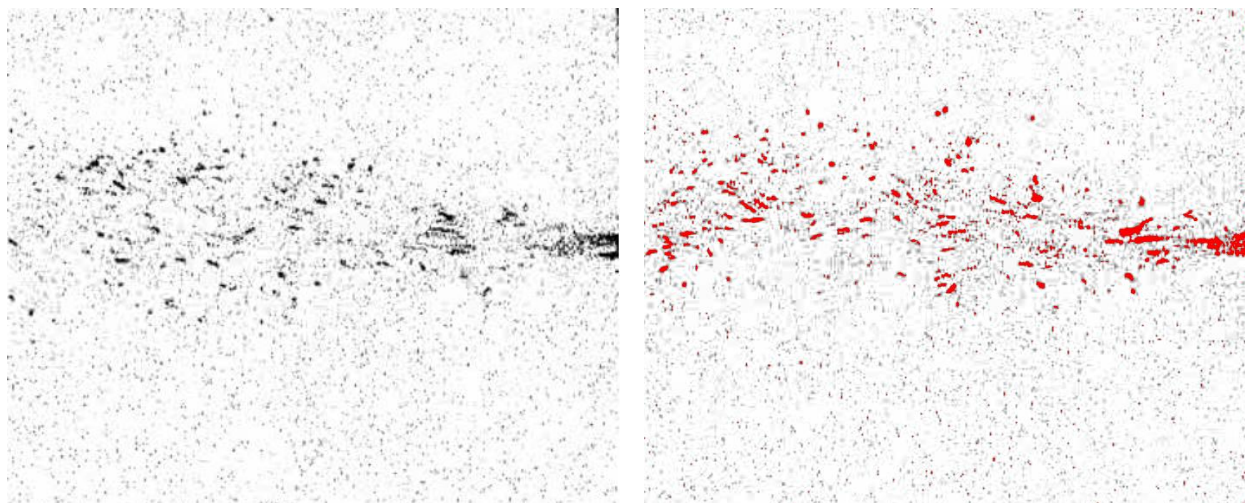
Pokud $B_{avg} < B_t$, pak každý pixel P_{XY} , jehož modrá složka $B_{XY} > B_t$ náleží bublině.



Snímek 32 Významné hodnoty modré barevné složky; kde:

- Červená oblast znázorňuje intenzity, které definují pixel bubliny
- B_{min} – Minimální hodnota modré složky ve snímku
- B_{max} – Maximální hodnota modré složky ve snímku
- B_{avg} – Průměrná hodnota modré složky ve snímku
- B_t – Prahová hodnota modré složky

Výsledek prahování konkrétní barevnou složkou je patrný na snímku 33b.



a) zdrojový snímek

b) snímek s vyznačenými pixely
heliových bublin (červeně)

Snímek 33 Prahování k označení pixelů heliových bublin

Zdroj: Odbor termomechaniky a techniky prostředí, VUT FSI [s.a.]

6.1.2 Výběr reprezentantů heliových bublin

Výsledkem této detekce bude seznam bodů, které představují potenciální heliové bubliny. Bublina je obecně spojitá oblast složená z několika pixelů. Postupuje se klasicky přes všechny pixely snímku, a jakmile je nalezen pixel bubliny, jsou vyhledány ostatní pixely dotyčné bubliny rekurzivním prohledáváním okolí prvního pixelu. Všechny zkoumané pixely jsou označeny, aby se jimi algoritmus v budoucnu již znovu nezabýval.

Po dokončení seznamu všech pixelů bubliny je z nich vybrán jeden, který ji bude reprezentovat. Protože jsou bubliny velmi malé, nehraje jeho výběr takovou roli. Reprezentantem může být například první, prostřední či poslední pixel v seznamu. Pokud byly během rekurzivního načítání pixelů bubliny evidovány také její extrémní souřadnice, pak z nich mohou být souřadnice reprezentanta vypočteny průměrováním:

$$X_r = \frac{X_{min} + X_{max}}{2} \quad (6.5)$$

$$Y_r = \frac{Y_{min} + Y_{max}}{2} \quad (6.6)$$

Potíží jsou oblasti, kde je velký shluk bublin (například v okolí vyústky). Tyto bubliny jsou tak rekurzivním algoritmem pojaty za jedinou bublinu, neboť je od sebe neoddělují pixely okolí. Pro další postup je to dokonce výhodné, neboť početně největší shluky bublin pomáhají určit zdroj proudění bublin. Za zdroj může být považovat největší shluk bublin, který zároveň leží na okraji proudu.

Shluky lze rozdělit na jednotlivé bubliny využitím té skutečnosti, že čím blíže středu bubliny se pixel nachází, tím vzdálenější je jeho příslušná barevná složka od stanoveného prahu. To lze vidět na snímku 34, který zachycuje zvětšený shluk o dvou bublinách.



Snímek 34 Shluk dvou bublin – detail

I v tomto případě je shluk rekurzivním algoritmem považován za jedinou bublinu, ale jsou v něm patrné dva lokálně extrémní pixely – dva pixely, které nemají v okolí žádný jiný ještě tmavší pixel. Každý z těchto pixelů bude reprezentantem jedné bubliny.

Poznámka: Rozřazování ostatních pixelů jednotlivým bublinám není pro další postup třeba. Lze však použít například jednoduché pravidlo, že konkrétní pixel náleží té bublině, jejíž reprezentant je k ní ze všech reprezentantů nejbližší.

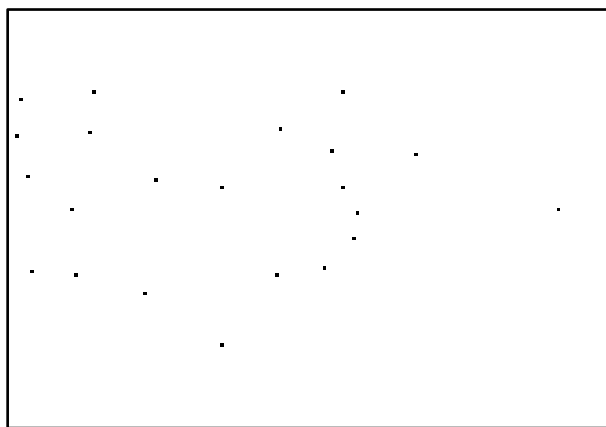
Výše uvedený postup tedy vybírá z obecných shluků bublin jednotlivé reprezentanty. Detekce pixelů bublin a výběr reprezentantů však lze spojit a výrazně zjednodušit: Snímek lze projít pouze jednou a reprezentanty v něm vybírat přímo. Pixel je reprezentantem, pokud splňuje obě tyto podmínky:

1. Barva pixelu odpovídá dle zvoleného prahu bublině
2. V okolí pixelu není jiný, jehož příslušná barevná složka by od prahu byla vzdálena více.

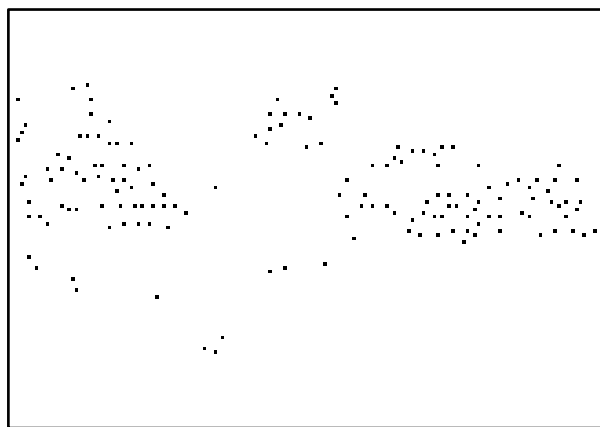
Snímek 35 zobrazuje okolí výustky, kde lze vidět stanovené reprezentanty pro shluky bublin (snímek 35b) i pro jednotlivé bubliny (snímek 35c).



a) zdrojový snímek



b) reprezentanti shluků bublin



c) reprezentanti jednotlivých bublin

Snímek 35 Výběr reprezentantů bublin v okolí kolem vyústky

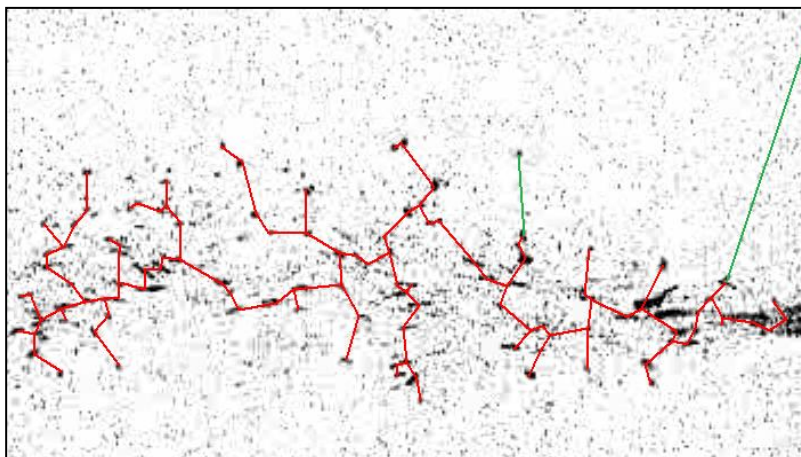
Možnost dělit barevné shluky na jednotlivé bubliny však nemusí být vždy žádoucí. O tvaru proudnice lze rozhodovat také jen dle reprezentantů shluků. Rozdělením shluků na bubliny dojde ke zpřesnění výsledku. Zejména v oblastech v okolí vyústky (snímek 35) je však bublin mnohonásobně více než shluků. Práce s tak velkým objemem dat může znamenat výrazný nárůst doby vyhodnocovacího procesu. Porovnáním snímků 35b a 35c lze vidět, že tvar proudnice by byl v obou případech velmi podobný. Z důvodů úspory času lze tedy převod shluků na reprezentanty jednotlivých bublin vynechat.

6.1.3 Separace neplatných reprezentantů

V předchozím postupu byla použitím konkrétních podmínek určena množina reprezentantů. Všichni tito reprezentanti splnili set určitých podmínek. Ne všichni však náleží hlavnímu proudu. Mohou představovat bubliny mimo proud, které však zůstaly ve snímku. Také mohou náležet jiným objektům, které nebyly před pořízením snímku odstraněny. Osamocená výrazná bublina mimo proud může velmi ovlivnit výsledný tvar proudnice. Účelem následujícího kroku je proto výběr pouze těch reprezentantů, které jsou v proudu.

Při práci se sekvencí by bylo možné považovat za bubliny proudu pouze ty, které výrazně změnily polohu. Pokud je na tomtéž místě (s malou tolerancí) ve dvou po sobě jdoucích snímcích nalezena bublina, pak tuto bublinu neunáší proud a lze ji pro následující postup ignorovat. Je-li však k dispozici jediný snímek proudění, je třeba uvažovat jiné postupy. Lze však využít toho, že bubliny nelétají osamoceně. Pokud bublina náleží proudu, v jejím okolí se budou vyskytovat další, které vyústku opustily v podobném čase a podobným směrem. Naopak bublina či množina bublin příliš vzdálených od ostatních pravděpodobně proudu nenáležejí.

Prvním krokem separace neplatných reprezentantů je sestrojení minimální kostry grafu těchto reprezentantů. Problém popisuje Jarník (1930) či Cheriton a Tarjan (1976). Kostra grafu je stromový graf, který propojuje všechny jeho vrcholy (reprezentanty). V tomto případě jde o kostru grafu, jejíž součet délek jednotlivých úsečků je minimální. K takové konstrukci lze využít například Kruskalův, Borůvkův či Jarníkův algoritmus. Snímek 36 zobrazuje minimální kostru grafu na množině reprezentantů shluků (viz 6.1.2).



Snímek 36 Minimální kostra grafu reprezentantů shluků heliových bublin
Zdroj: Šťastný, Richter, 2016

Poznámka: Zelené úsečky nesplňují podmínku pro velikost úsečky dané vzorcem 6.7.

Kostra grafu bude použita ke statistickému vyhodnocení, pomocí něhož bude určeno, zda není bublina či více bublin příliš vzdálených od ostatních. Pro tento účel je třeba určit následující parametry:

- 1) Počátek proudění E_{XY} – Většinou v místě největšího shluku bublin. Obecněji v místě největšího shluku bublin, který leží na konvexní obálce nezredukovaného seznamu reprezentantů (algoritmus bude uveden v kapitole 6.2).
- 2) Průměrná velikost úseček v minimální kostře grafu L_{avg} .
- 3) Středů všech úseček v kostře.
- 4) Největší vzdálenost d_{max} středu úsečky od počátku proudu E_{XY} .

Podmínka, která určí, které úsečky jsou platné, je pak stanovena jako kombinace výše zmíněných faktorů:

$$L \leq \left(k + \frac{d}{d_{max}} q \right) L_{avg} \quad (6.7)$$

Kde: L – délka vyšetřované úsečky

d – vzdálenost středu vyšetřované úsečky od počátku proudění

k – statický koeficient

q – dynamický koeficient

Pokud střed úsečky (spojnice dvou bublin) leží přímo v počátku proudění, pak je jí povolena maximální délka $k \times L_{avg}$. S rostoucí vzdáleností d od počátku proudění je však možné tolerovat i větší vzdálenosti mezi jednotlivými bublinami, což závisí na dynamickém koeficientu q . Oba koeficienty k i q jsou však stanoveny experimentálně. Jejich nastavení se může mírně lišit s ohledem na množství bublin v proudnici a také s ohledem na to, jestli se pracuje s reprezentanty shluků či s reprezentanty jednotlivých bublin. Pro snímek 36, kde se operuje s reprezentanty shluků, je to $k = 1.5$, $q = 1.5$.

Úsečky u počátku proudu mohou dosahovat maximálně 1.5násobku průměrné velikosti úsečky. Nejvzdálenější úsečka od počátku pak může mít oproti průměru až trojnásobnou velikost. Kvůli

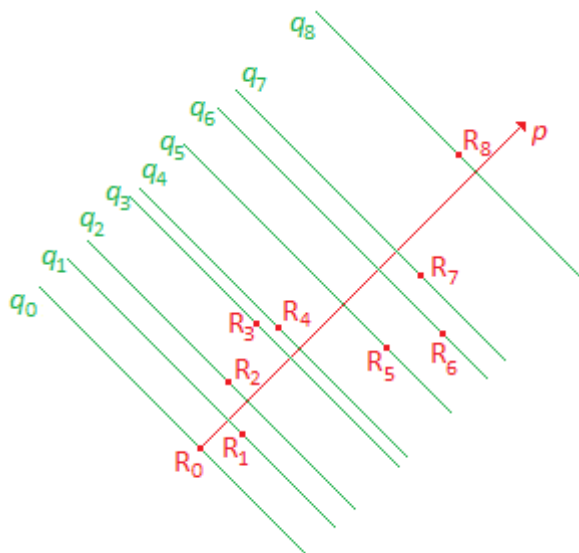
nesplnění podmínky (6.7) byly ve snímku 36 zneplatněny dvě úsečky (označené zeleně). Reprezentanty či skupiny reprezentantů, které jsou s proudem spojeny těmito úsečkami, tedy nebudeme do proudu uvažovat. Úsečka spojuje obecně dvě množiny reprezentantů. S jejím odstraněním budou odstraněni všichni reprezentanti jedné z těchto množin. Neplatnou množinu reprezentantů lze vybrat jako například tu méně početnou, nebo lépe jako tu, která neobsahuje počátek proudění.

6.1.4 Seřazení množiny reprezentantů

Obecně není nutné detekované platné reprezentanty k určení jejich obálky seřadit. Pro účely této práce je však výhodné jejich seřazení provést, čehož bude využito při rekonstrukci hranice z chromozomu při sestrojování nekonvexní obálky genetických algoritmů (viz kapitola 6.3.2). Kvůli zmíněnému kroku je třeba reprezentanty seřadit následujícím postupem:

1. Je určen hlavní směr proudu úhlem φ – průměr sklonů spojnic počátku proudění s ostatními platnými reprezentanty.
2. Je určena hlavní přímka proudu p se sklonem φ , procházející reprezentantem počátku.
3. Každému reprezentantovi R_i bude náležet přímka q_i , která jím prochází, a přitom je kolmá na hlavní směr proudu.
4. Přímky q_i seřadíme vzestupně podle toho, jak je od reprezentanta počátku vzdálený jejich průnik s hlavní přímkou p . Spolu s přímkami řadíme i jim náležící reprezentanty.

Snímek 37 znázorňuje proces i výsledek seřazení reprezentantů.

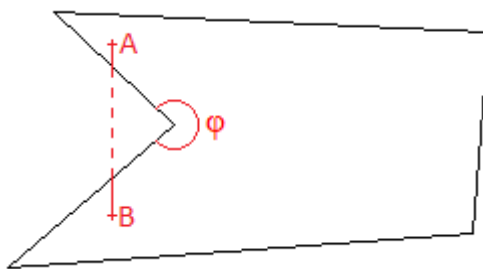


Snímek 37 Reprezentanti R_i seřazení podle vzdálenosti jim příslušících přímek q_i (kolmých na hlavní směr proudění) od počátku proudění

6.2 URČENÍ KONVEXNÍ OBÁLKY PROUDNICE

Obálkou proudnice se rozumí obálka, která ohraničuje oblast, v níž se nachází všichni reprezentanti. Pokud je taková obálka konvexní, pak všechny její vnitřní úhly jsou konvexní ($< \pi$).

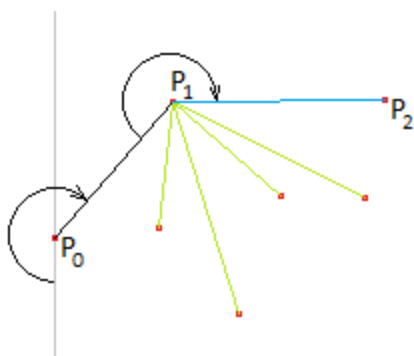
To lze definovat i tak, že jakákoliv úsečka spojující libovolné dva body uvnitř obálky leží celá uvnitř této obálky.



Snímek 38 Nekonvexní obálka

Na snímku 38 lze vidět nekonvexní obálku. Úhel φ je vnitřní konkávní úhel. Body A a B leží oba uvnitř obálky, avšak úsečka AB , která je spojuje, zasahuje i mimo tuto obálku (čárkovaná oblast).

Problém určení konvexní obálky lze vyřešit pomocí postupu Jarvis scan (Jarvis, 1973). Algoritmus je efektivní a spočívá v postupném přidávání hraničních bodů (reprezentantů) do hranice.



Snímek 39 Princip tvorby konvexní obálky

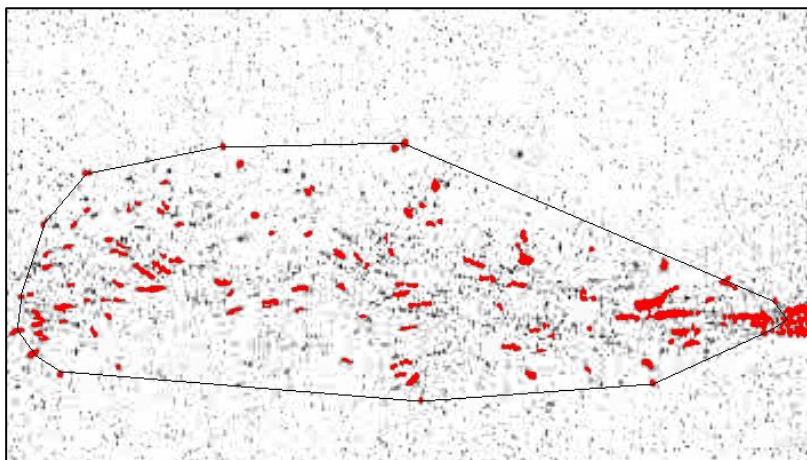
Princip algoritmu demonstruje snímek 39, kde je ukázáno, jak zařadit bod P_{i+1} (konkrétně P_2) do křivky obálky. Aktuálním bodem je bod P_i (konkrétně P_1). Tento a předchozí bod P_{i-1} tvoří polopřímku $P_i P_{i-1}$. Polopřímka je rotována kolem bodu P_i po směru hodinových ručiček (směr rotace zvolen před zahájením algoritmu). První z bodů, který je při rotaci protnut, bude bodem P_{i+1} . Lze říci, že jsou procházeny všechny body P_j (různé od P_i) tak, aby byl nalezen nejmenší z úhlů $P_{i-1} P_i P_j$. Dotyčný bod P_j se pak stává novým bodem obálky P_{i+1} .

Poznámka: Z dvojice možných úhlů $P_{i-1} P_i P_j$ je uvažován ten, jehož velikost je v rozmezí $< \pi, 2\pi$.

Konkrétně na snímku 39 je nový bod P_2 vybrán proto, že pro všechny ostatní body by byl nový úhel (konkávní nebo přímý) na obálce větší. K vybrání bodu P_{i+1} je tedy třeba dvojice předchozích bodů P_i a P_{i-1} . Vlastnímu algoritmu tak musí předcházet proces inicializace. V inicializační fázi je vybrán bod P_0 , který s jistotou leží na hranici. Může to být například bod s nejnižší souřadnicí X (v případě více takových bodů z nich vybereme ten s nejnižší souřadnicí Y). Jemu je dále zvolen fiktivní předchůdce P_{-1} . Tím může být libovolný bod různý od P_0 , který má souřadnici X menší nebo rovnou souřadnici X bodu P_0 . Algoritmus začíná v bodě P_0 a obálku neustále rozšiřuje o nové body

P_{i+1} , dokud nebude jako nový bod P_{i+1} označen právě počáteční bod P_0 . Tím se obálka uzavře a všichni reprezentanti budou ležet buď uvnitř této obálky nebo na hranici.

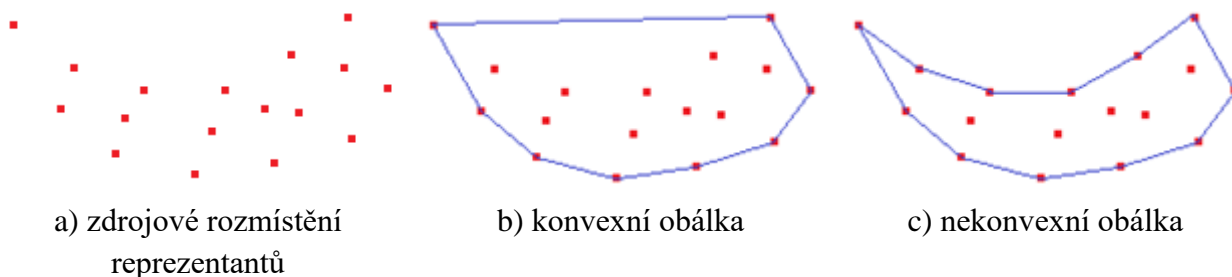
Snímek 40 zobrazuje detekovanou konvexní obálku ve skutečném snímku proudění heliových bublin, kdy algoritmus byl aplikován na množinu reprezentantů shluků heliových bublin.



Snímek 40 Konvexní obálka reprezentantů shluků heliových bublin

6.3 NEKONVEXNÍ OBÁLKA PROUDNICE

Definice nekonvexní obálky byla již popsána v kapitole 6.2. Snímek 41 zachycuje porovnání hranice téhož proudu – téže množiny reprezentantů, přičemž snímek 41b zobrazuje konvexní a snímek 41c nekonvexní ohrazení.



Snímek 41 Porovnání konvexní a nekonvexní hranice
Zdroj: Šťastný, Richter, 2016

Z porovnání je patrné, že nekonvexní obálka zachycuje proudění věrněji právě proto, že ji neomezuje podmínka konvexnosti oblasti. Postup sestavení nekonvexní obálky však nelze jednoznačně definovat. Jednou možností by mohlo být zlepšování hranice postupným přidáváním dalších bodů do původně konvexní hranice, pokud se tím hranice vylepší. To může být velmi obtížně definovatelné. Zároveň se může stát, že se přidáním konkrétního bodu hranice nezlepší, ale je nutné ho i tak přidat, aby bylo docíleno ještě kvalitnější hranice. Další možností je postupné propojování bodů úsečkami, například od nejkratší možné. Postupně by se tak konstruoval graf. Ten by ve výsledku musel být spojitý a cyklický a musely by se z něj následně odstranit vnitřní spojnice

uvnitř obálky. Tyto podmínky mohou být velmi obtížně definovatelné, a i přes to nezaručují kvalitní výslednou hranici, protože ji nehodnotí jako celek.

Základní otázkou zůstává, jaké body do hranice zahrnout (jak je zkombinovat), aby byla co nejvěrohodnější. Pro jednu variantu hranice může být rozšíření o konkrétní bod výhodné, pro jinou nikoliv. Nutnost porovnávat hranici jako celek a potřebu porovnávání různých variant hranice lze zkombinovat využitím genetických algoritmů.

6.3.1 Použití genetického algoritmu při konstrukci nekonvexní obálky

Použití genetického algoritmu popisuje Šťastný a Richter (2016). Každý jedinec v populaci bude představovat jedno konkrétní řešení ohraničení pole reprezentantů. Pole reprezentantů R_0-R_n je neměnné (stejně seřazené) pro všechny jedince po celou dobu vyhodnocování. Velikost chromozomu odpovídá velikosti pole reprezentantů. Všechny geny chromozomu obsahují binární hodnotu. Pokud má gen G_i hodnotu 1, pak reprezentant R_i je prvkem hranice, pokud má G_i hodnotu 0, pak R_i hranici nenáleží.

Hranice se mezi sebou různě kombinují, čímž mimo jiných vznikají i jedinci, kteří mají dobré vlastnosti obou (obecně všech) svých rodičů. Dobrá vlastnost v tomto případě znamená prezenci konkrétního reprezentanta či skupiny reprezentantů v hranici. Jedinec (varianta hranice) je jednoznačně určen svým chromozomem. Jednotlivé jeho geny definují, které body náleží hranici a které nikoliv. Tyto body je však nutné správně seřadit tak, aby jejich postupný průchod vytvořil hranici (na závěr je spojen poslední a první bod hranice). Jakmile je hranice kompletní, zbývá ji ohodnotit tzv. fitness funkcí. Díky ní získáme kvalitativní fitness hodnotu, která bude určovat šanci, s jakou se dotyčný jedinec bude křížit, aby předal část svého chromozomu na další generaci.

Před použitím genetického algoritmu je třeba zvolit jeho parametry (viz kapitola 3.6):

Velikost populace

Odvíjí se od počtu reprezentantů n . Jestliže gen pro každého reprezentanta může nabývat dvou hodnot, pak maximální počet kombinací hranice je 2^n . Během testování byl počet populace v genetickém algoritmu přibližně roven počtu reprezentantů bublin, avšak vždy byl v rozmezí 50-200. Tedy lze doporučit:

Tabulka 4 Doporučená velikost populace pro daný počet reprezentantů

Počet reprezentantů n	Velikost populace q
$\langle 0, 49 \rangle$	50
$\langle 50, 200 \rangle$	n
$\langle 201, \infty \rangle$	200

Výběr rodičů

V práci je použitý model dvou rodičů jednoho potomka. Není však dokázáno ani vyvráceno, že je to model nejlepší. Rodiče jsou vybíráni ruletovým způsobem, a to především proto, že tento způsob je nejméně náročný na strojový čas.

Počet dětí v populační obměně

Je žádoucí, aby se v rámci populační obměny nahradilo velké množství populace. Zároveň je však nežádoucí, aby kvalitní potomci zanikali. V práci se proto populační obměna týká právě poloviny populace – polovina populace bude zachována a polovina bude nahrazena potomky v nové generaci. Jedinci, kteří jsou při tom vylučováni z populace, jsou vybíráni dle fitness hodnoty od nejméně kvalitních.

Způsob křížení

Protože množina reprezentantů není seřazena podle toho, jak budou v hranici následovat za sebou (ani to nelze, protože množina musí být seřazena stejně pro všechny možné hranice), nehraje roli, jestli použijeme bodové křížení s jedním dělicím bodem, vícebodové křížení s několika dělicími body, či budeme určovat rodiče pro každý gen potomka zvlášť. Z hlediska procesního času i složitosti algoritmu jsou první a třetí jmenovaná možnost takřka ekvivalentní.

Mutace

Mutaci je vhodné nastavit v závislosti na počtu genů n (a tedy i reprezentantů). V práci je pro tento účel vytvořena pravděpodobnostní tabulka, sloužící k určení počtu mutací. Maximální počet mutací q stanovíme jako celočíselný podíl počtu genů (omezený intervalem $<50, 200>$) a 50:

$$q = \min[200, \max(50, n)] \text{ div } 50 \quad (6.8)$$

Pravděpodobnost, že k mutaci nedojde, je vždy 70 %. Zbýlých 30 % se proporcionálně rozdělí mezi pravděpodobnosti počtu mutací, jak ukazuje tabulka 5:

Tabulka 5 Pravděpodobnost počtu mutací v závislosti na počtu genů

Počet genů	<100	100-149	150-199	>=200
Q	1	2	3	4
Pravděpodobnost 0 mutací	70 %	70 %	70 %	70 %
Pravděpodobnost 1 mutací	30 %	15 %	10 %	7,5 %
Pravděpodobnost 2 mutací		15 %	10 %	7,5 %
Pravděpodobnost 3 mutací			10 %	7,5 %
Pravděpodobnost 4 mutací				7,5 %

Tedy například při vytváření jedince o méně než 100 genech je 30% šance, že dojde právě k jedné mutaci. Pomocí tabulky a generátoru náhodného čísla tak bylo určeno, kolik genů nového jedince by mělo zmutovat, v našem případě změnit hodnotu na opačnou.

Ukončení genetického algoritmu

Pro účely tohoto výzkumu je použito maximálního počtu generačních obměn beze změny nejlepšího jedince.

6.3.2 Sestrojení hranice dle chromozomu

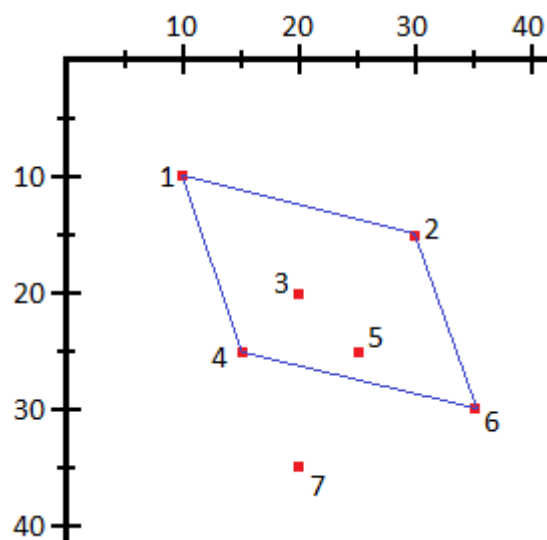
Každý jedinec je přesně daný svým chromozomem. Jak bylo řečeno, jednotlivé geny popisují, zda příslušný reprezentant náleží hranici či nikoliv. Do procesu sestrojování hranice proto vstupují pouze ti reprezentanti, jejichž geny mají hodnotu 1. V této fázi tak není třeba řešit, jestli všichni reprezentanti leží uvnitř obálky, nebo jsou nějací vně. V tomto bodě je třeba body hranice seřadit tak, jak v ní následují po sobě. Tabulka 6 je příkladem počátečního stavu, tedy všech reprezentantů a hodnot jim příslušných genů. Tabulka 7 pak zachycuje požadovaný výsledek sestrojování hranice. Graficky jsou reprezentanti i sestrojená hranice znázorněni na snímku 42.

Tabulka 6 Příklad množiny reprezentantů s příslušnými geny

Index reprezentanta i genu	Souřadnice X	Souřadnice Y	Hodnota genu
1	10	10	1
2	30	15	1
3	20	20	0
4	15	25	1
5	25	25	0
6	35	30	1
7	20	35	0

Tabulka 7 Správně seřazené hraniční body (požadovaný výstup)

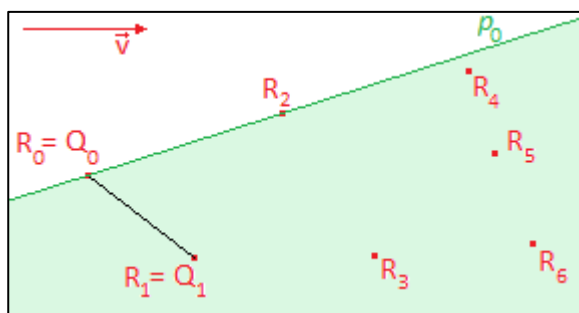
Pořadí	Původní index	Souřadnice X	Souřadnice Y	Hodnota genu
1	1	10	10	1
2	4	15	25	1
3	6	35	30	1
4	2	30	15	1



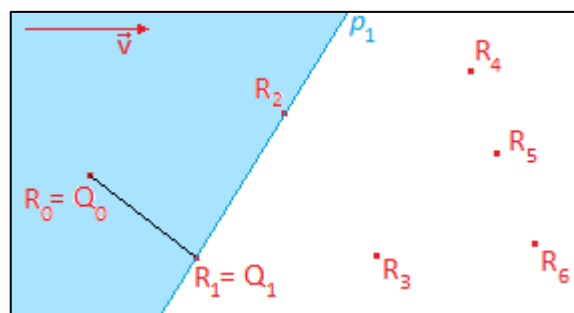
Snímek 42 Sestrojená hranice nad danou množinou bodů pro chromozom 1101010
Zdroj: Šťastný, Richter 2016

Na algoritmus seřazení hraničních bodů jsou kladeny značné nároky na co nejmenší procesní čas, protože algoritmus bude vykonán v rámci vytváření každého jedince. Z důvodu úspory času tento proces vyžaduje seřazení množiny reprezentantů (viz kapitola 6.1.4) ještě před předáním reprezentantů genetickému algoritmu.

K jednotlivým reprezentantům budeme přistupovat postupně, jak jsou seřazené. Reprezentanty s příslušnou hodnotou genu 0 pouze přeskočíme. Reprezentanty s příslušnou hodnotou genu 1 budeme postupně přiřazovat do hranice. V inicializační fázi spojíme první dva reprezentanty hranice úsečkou. Oba body označíme jako koncové body dosavadní hranice, tedy Q_0 a Q_1 . Pro každého následujícího reprezentanta hranice je pak rozhodováno, zda má být úsečkou spojen s bodem Q_0 (a stát se tak novým koncovým bodem Q_0), nebo s bodem Q_1 (a stát se tak novým koncovým bodem Q_1). Pro potřeby rozhodnutí, ke kterému z konců se má vyšetřovaný bod připojit, je třeba geometrické konstrukce, kterou demonstruje snímek 43, kde se konkrétně rozhoduje, ke kterému z konců by se měl připojit bod R_2 . Tuto konstrukci budeme dále nazývat „rozhodovací geometrická konstrukce“.



a) geometrická konstrukce pro koncový bod Q_0



b) geometrická konstrukce pro koncový bod Q_1

Snímek 43 Volba konce pro připojení vyšetřovaného bodu; kde:

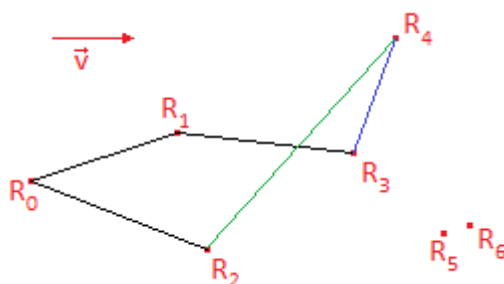
R_i – reprezentanti

p_j – přímky dané příslušným koncovým bodem Q_j a vyšetřovaným bodem R_2

\vec{v} – směrový vektor proudění (sloužil pro seřazení reprezentantů)

Snímek 43a prezentuje rozhodovací geometrickou konstrukci pro bod Q_0 . Body Q_0 a R_2 (vyšetřovaný) určí přímku p_0 . Tato přímka a druhý z koncových bodů Q_1 definují polorovinu p_0Q_1 (na snímku 43a zeleně znázorněná). Obdobně postupujeme pro bod Q_1 : body Q_1R_2 určí polopřímku p_1 a ta spolu s bodem Q_0 určí polopřímku p_1Q_0 (na snímku 43b modře znázorněná). Následně v obou polorovinách spočítáme počty doposud nezařazených reprezentantů. Vyšetřovaný bod bude přidělen k tomu konci, jehož zkonstruovaná polorovina obsahuje více dosud nezařazených reprezentantů. V tomto případě obsahuje polorovina p_0Q_1 4 reprezentanty a polorovina p_1Q_0 neobsahuje žádného. Bod R_2 se úsečkou spojí s bodem Q_0 a sám se stane novým koncovým bodem Q_0 .

Během geometrické konstrukce je nutné ověřit, že úsečky (nikoliv přímky) Q_0R_i a Q_1R_i neprotínají již sestrojenou hranici. Pokud by některá hranici protínala, je automaticky neplatná a bod R_i se připojí k druhé z nich. Snímek 44 ukazuje, jak může takový jev nastat.



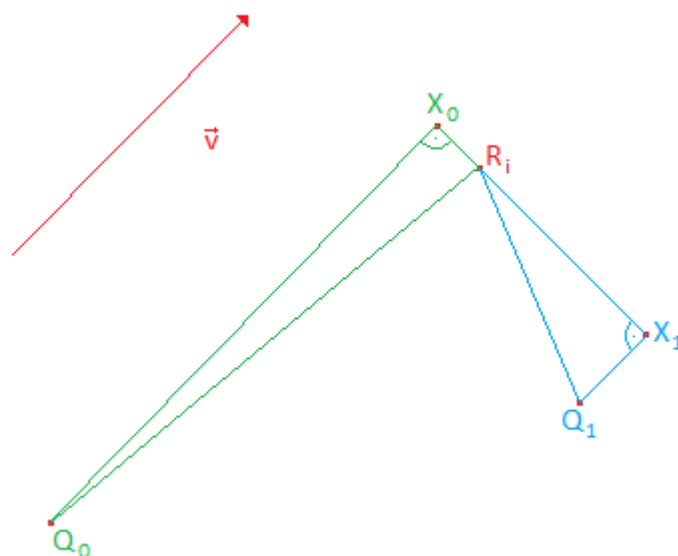
Snímek 44 Průnik hranice při rozhodovací geometrické konstrukci

Bod R_3 byl spojen s koncem R_1 , neboť v polorovině $R_1R_3R_2$ leží body R_5 a R_6 , zatímco v polorovině $R_2R_3R_1$ ležel pouze bod R_4 . Když pak v dalším kroku k hranici přiřazován bod R_4 , úsečka R_2R_4 protíná již existující úsečku hranice R_1R_3 . Proto ji nelze uvažovat a bod R_4 bude automaticky spojen s koncem R_3 .

Může také nastat případ, kdy obě poloroviny obsahují stejný počet nezařazených reprezentantů. V takovém případě jsou sestrojeny pravoúhlé trojúhelníky:

- $Q_0R_iX_0$, s pravým úhlem při vrcholu X_0 a stranou Q_0X_0 rovnoběžnou s vektorem směru proudu \mathbf{v} . Tedy strana R_iX_0 bude kolmá na směr proudu.
- $Q_1R_iX_1$, s pravým úhlem při vrcholu X_1 a stranou Q_1X_1 rovnoběžnou s vektorem směru proudu \mathbf{v} . Tedy strana R_iX_1 bude kolmá na směr proudu.

Pokud je úsečka R_iX_0 menší než R_iX_1 , pak se bod R_i přiřadí ke konci Q_0 , v opačném případě ke konci Q_1 . Postup demonstruje snímek 45. Protože v tomto případě je úsečka X_0R_i kratší než úsečka X_1R_i , bod R_i bude spojen s koncem Q_0 a stane se novým koncovým bodem Q_0 .



Snímek 45 Hledání bližšího konce ve směru kolmém na směr proudění; kde:

Q_0, Q_1 – koncové body doposud sestrojené hranice

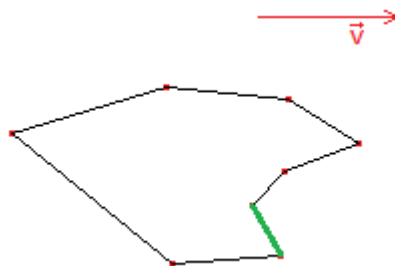
R_i – vyšetřovaný bod

X_0, X_1 – pomocné body v trojúhelnících, při nichž je pravý úhel

\mathbf{v} – směrový vektor proudu, je vektorem úseček Q_0X_0 a Q_1X_1

Poté, co jsou do hranice zapojeni všichni hraniční reprezentanti, zbývá úsečkou spojit koncové body q_0 a q_1 , čímž se obálka uzavře. Uvedený algoritmus je velmi rychlý. Procesní čas je lineárně závislý na počtu reprezentantů (kromě sčítání bodů v polorovinách, které je však elementární). Byly vykovávány pokusy o vytvoření grafu různých variant seřazení bodu, ale takový úkol je kvůli exponenciální závislosti na počtu bodů nepoužitelný.

Výhodou algoritmu rozhodovací geometrické konstrukce je tedy rychlost seřazení hranice. Nevýhodou může být postupné řazení bodů. Na snímku 46 lze vidět hranici, kterou nikdy nelze algoritmem dosáhnout. Při postupu po hranici od počátku proudu až po nejvzdálenější bod přechodem na každý další bod hranice roste vzdálenost od počátku (uvažuje se pouze vzdálenost rovnoběžná se směrem proudu). Hranice na snímku 46 však obsahuje takzvanou návratovou úsečku. Tedy výše uvedené pravidlo nesplňuje. Takové proudy sice existovat mohou, například kvůli turbulencím. Protože je však hranice detekována pouze v jediném snímku, odhalit turbulenci by bylo velmi obtížné i při použití komplexnějších a časově mnohem náročnějších algoritmů.



Snímek 46 Příklad hranice s návratovou úsečkou (zelená)

6.3.3 Fitness funkce

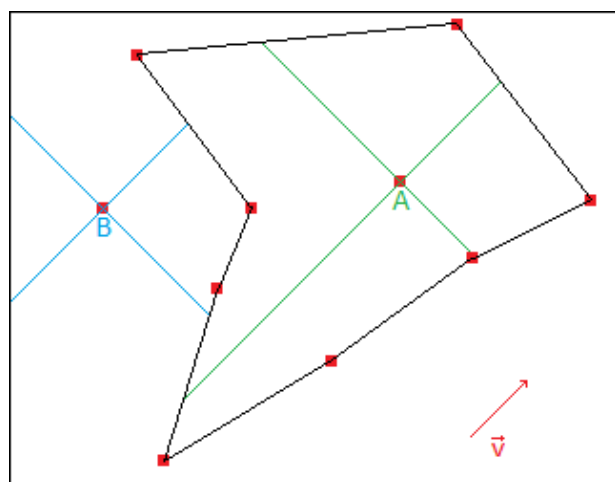
Jedná se o proceduru, která komplexně ohodnotí kvalitu daného jedince reálným číslem – tzv. fitness hodnotou. Fitness hodnota jednak udává, který jedinec je nejlepší, jednak také zvýhodňuje kvalitnější jedince při výběru rodičů pro další křížení. V tomto případě bude fitness funkce hodnotit kvalitu hranice sestavené z chromozomu (viz předchozí kapitola 6.3.2). Podobně jako při sestavování hranice, i zde platí značné nároky na rychlost celého procesu, protože i fitness funkce bude volána při inicializaci každého jedince.

Zkonstruovaná hranice může být posuzována celou řadou kritérií. Každé kritérium však znamená strojový čas navíc. Je třeba je zúžit na nejnutnější, které však řešení budou správně usměrňovat. Zároveň musí být co nejméně náročné na dobu evaluace.

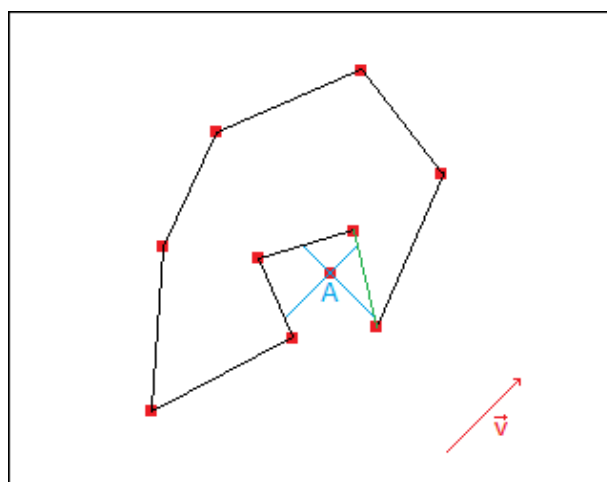
Počet bodů vně obálky

Hranice proudu by měla obsáhnout ideálně všechny platné reprezentanty. Z platných reprezentantů postačí kontrolovat ty, které mají hodnotu příslušného chromozomu 0. Zbylé leží na hranici.

Každý reprezentant, který leží vně, musí snížit kvalitu hranice. Obecně platí, že pokud existuje cesta od bodu k hranici snímku, pak bod leží vně obálky. Cesta obecně neznamená přímku, ale jakoukoliv křivku. Lze však využít skutečnosti, že snímek neobsahuje návratovou úsečku. Proto stačí od bodu postupovat přímo čtyřmi různými směry: ve směru proudu, proti směru proudu a dvěma směry kolmými na směr proudu. Směry demonstruje snímek 47a. Pokud alespoň v jednom ze směrů dorazíme od bodu až k hranici snímku, aniž bychom narazili na obálku, dotyčný bod je vně obálky. Snímek 47b ukazuje hranici, pro kterou by toto kritérium nešlo použít a bylo by nutné hledat cestu komplexním (a časově daleko náročnějším) algoritmem. Hranici na snímku 47b však nelze algoritmem zkonstruovat, protože tento neumožňuje prezenci návratové úsečky (viz kapitola 6.3.2).



a) hranice dle algoritmu 6.3.2



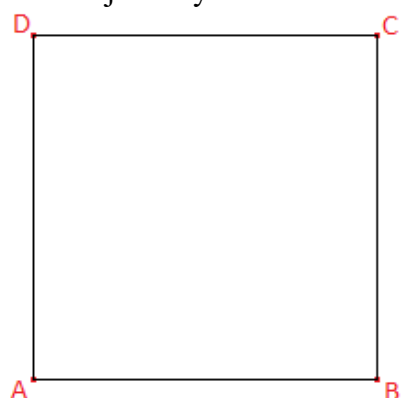
b) teoretická hranice s návratovou úsečkou (zeleně)

Snímek 47 Hledání cesty z bodu na hranici

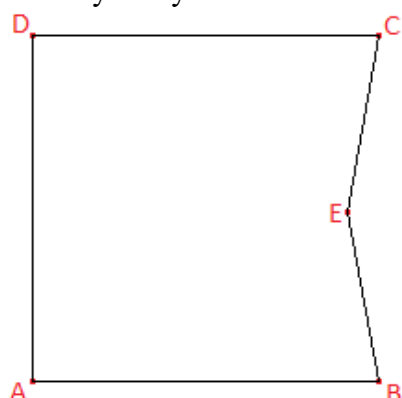
Je zřejmé, že jediný bod mimo obálku pro malý počet bodů představuje větší nepřesnost než jediný bod mimo obálku pro velký počet bodů. Spíše než počet bodů bude více vypovídajícím faktorem procento bodů vně obálky.

Plynulost hranice

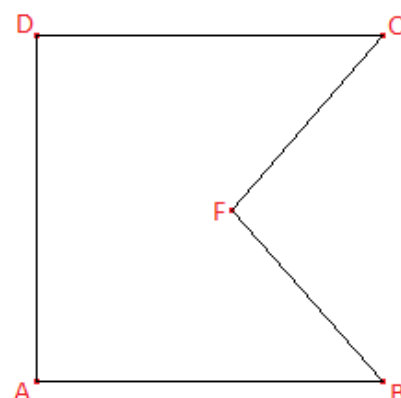
Obecně je sestavená hranice souborem úseček, tedy po částech přímá čára. Zlomy (změny úhlů) mezi jednotlivými úsečkami by však neměly být příliš ostré. Na druhou stranu by hranice měla pojmut co nejvíce bodů, aby působila co nejvěrohodněji. Jsou to tedy dva protichůdné požadavky, a to co nejmenší součet změn úhlů v hranici a co největší počet bodů hranice. Obě kritéria lze spojit v jedno kontrolováním průměrné změny úhlu hranice. Jako příklad poslouží čtvercová oblast, tedy všechny změny úhlů mají hodnotu 0.5π a stejná je i hodnota průměrné změny úhlu. Pokud bude do hranice přibrán další bod, zvětší se tím počet úhlů hranice o jeden. Spolu s tím do sumy změn úhlů přibude jeden a dva jemu sousední budou ovlivněny. Pokud uvažovaný bod leží blízko hranice, je rozšíření do něj žádoucí, neboť nárůst sumy úhlů bude vykompenzován nárůstem počtu úhlů. Pro vzdálenější body už toto rozšíření může být nevýhodné.



a) základní snímek



b) rozšíření o nevhodný bod



c) rozšíření o vhodný bod

Snímek 48 Rozšíření čtvercové obálky

Popsaný příklad demonstruje snímek 48. Snímek 48b představuje vhodné rozšíření. Přibyla nová změna úhlu ($\pi - CEB$), což však představuje jen velmi malý úhel. Dva další úhly jsou ovlivněné. Místo úhlu DCB je v hranici nyní úhel DCE a místo úhlu ABC je úhel ABE . Zmiňovanými změnami se celková změna úhlů podél hranice sice mírně zhorší, přesto díky zvětšení počtu úhlů znamená zlepšení průměrné změny úhlu v hranici (nyní 0.42π tedy zlepšení plynulosti hranice). Rozšíření o bod E je tedy žádoucí.

Naproti tomu rozšíření o bod F (snímek 48c) znamená velké změny směru ABF , BFC i FCD . Průměrná změna směru je pak 0.6π , což znamená z pohledu plynulosti hranice zhoršení.

Je také otázkou, zda operovat přímo se změnami úhlů, či je modifikovat například mocninou, logaritmicky či exponenciálně.

Sestavení funkčního předpisu

Je zřejmé, že ohraničená oblast by měla obsahovat všechny reprezentanty bublin. Pro účely genetického algoritmu však není vhodné tento kvalitativní faktor zcela upřednostnit. Mohou existovat varianty, kdy je plynulost hranice v jedné části velmi kvalitní. V jiné části hranice však mohou existovat reprezentanti, které obálka nepojmula. Pokud by bylo pravidlo o počtu reprezentantů uvnitř obálky zcela upřednostněno, neměl by takový jedinec velkou šanci na reprodukci, přestože jeho část je velmi kvalitní.

Je tedy třeba hledat kompromisní nastavení obou faktorů spolu s nastavením konkrétních dílčích operátorů v jednotlivých výpočtech. Opakovaným testováním byla stanovena komplexní fitness funkce, pro výpočet fitness hodnoty i -tého jedince:

$$F_i = k_0 + \frac{1}{k_1 \cdot \frac{o}{n} + k_2 \cdot \ln\left(\frac{\sum e^{\varphi_i}}{\sum i}\right)} \quad (6.9)$$

Kde

o – počet bodů vně hranice

n – počet všech bodů

$\sum i$ – počet bodů hranice

$k_0 = 0.1$ – minimální fitness hodnota, aby měli i nekvalitní jedinci šanci se křížit.

$k_1 = 12$ – faktor postihující kritérium počtu bodů vně oblasti.

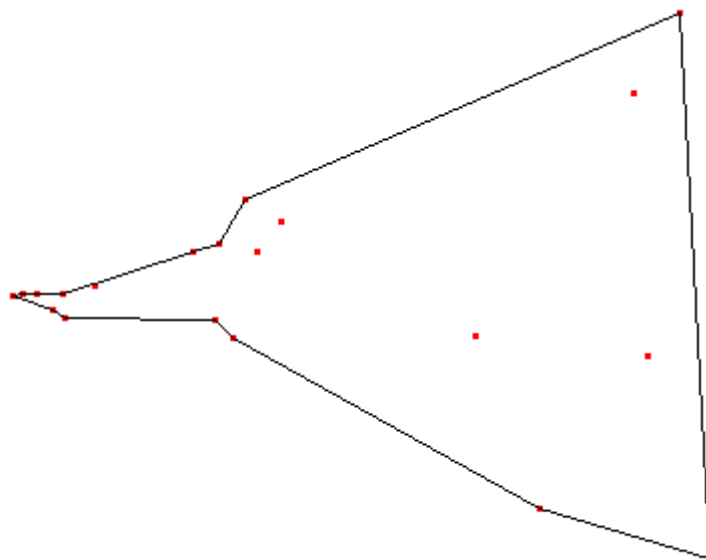
$k_2 = 2.2$ – faktor postihující kritérium plynulosti hranice

Poznámka: před výpočtem fitness hodnoty je třeba ošetřit případná dělení 0. Pokud by mělo nějaké nastat, bude fitness hodnota odpovídat přímo hodnotě k_0 .

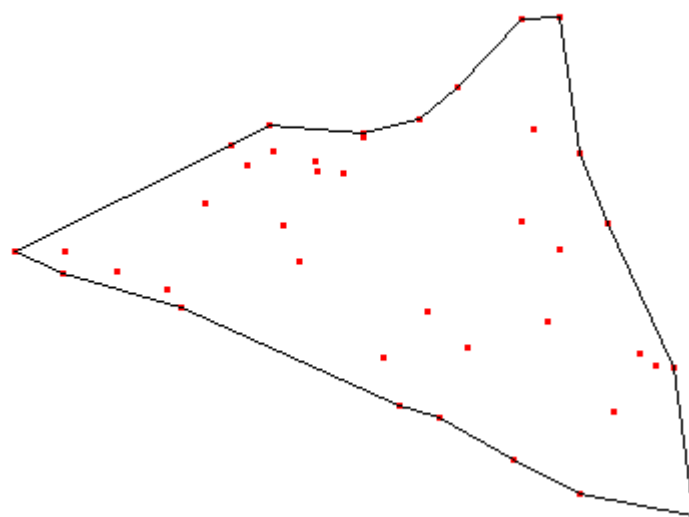
Změny úhlů při ohodnocení plynulosti hranice jsou uvažovány exponenciálně a až výsledný průměr je převedený přirozeným logaritmem zpět do původního měřítka. Tímto opatřením vyšší úhly ještě více negativně ovlivňují fitness hodnotu. Hodnoty k_1 a k_2 slouží k upřednostnění jednoho či druhého faktoru. Výsledky obou kritérií jsou ve jmenovateli, protože obě představují číselnou hodnotu chyby či ohodnocení nežádoucího jevu.

6.3.4 Průběh genetického algoritmu

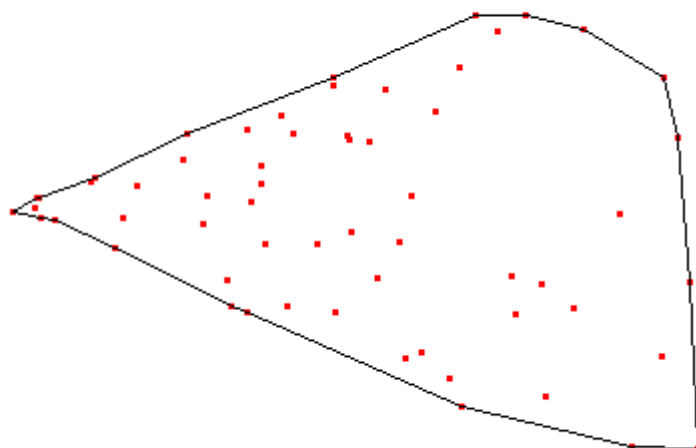
Genetický algoritmus je tedy nastaven a je stanoven proces vytvoření a ohodnocení jedince. Algoritmus tak může být testován, a to pro náhodné množiny o 20, 40, 60 a 120 bodech. 120 bodů je přibližný počet reprezentantů ve snímku 33a (reálný). Body byly vždy generovány náhodně v trojúhelníkové oblasti. Na snímcích 49, 50, 51 a 52 lze vidět výsledky jedné z variant pro každý z uvažovaných počtů bodů.



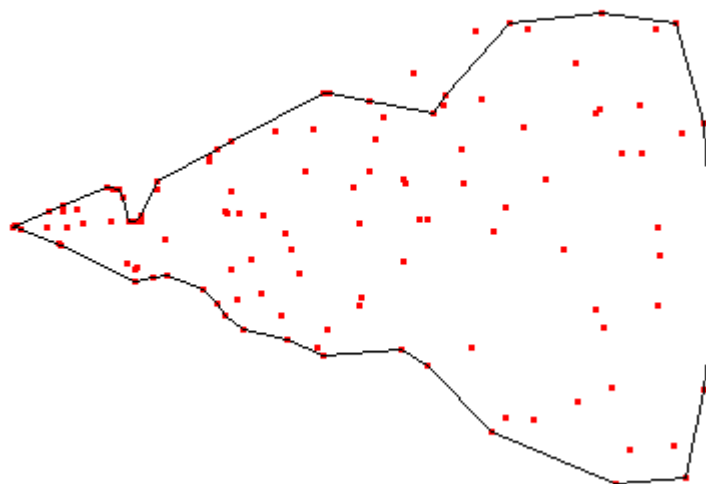
Snímek 49 Výsledek genetického algoritmu pro 20 bodů



Snímek 50 Výsledek genetického algoritmu pro 40 bodů



Snímek 51 Výsledek genetického algoritmu pro 60 bodů



Snímek 52 Výsledek genetického algoritmu pro 120 bodů

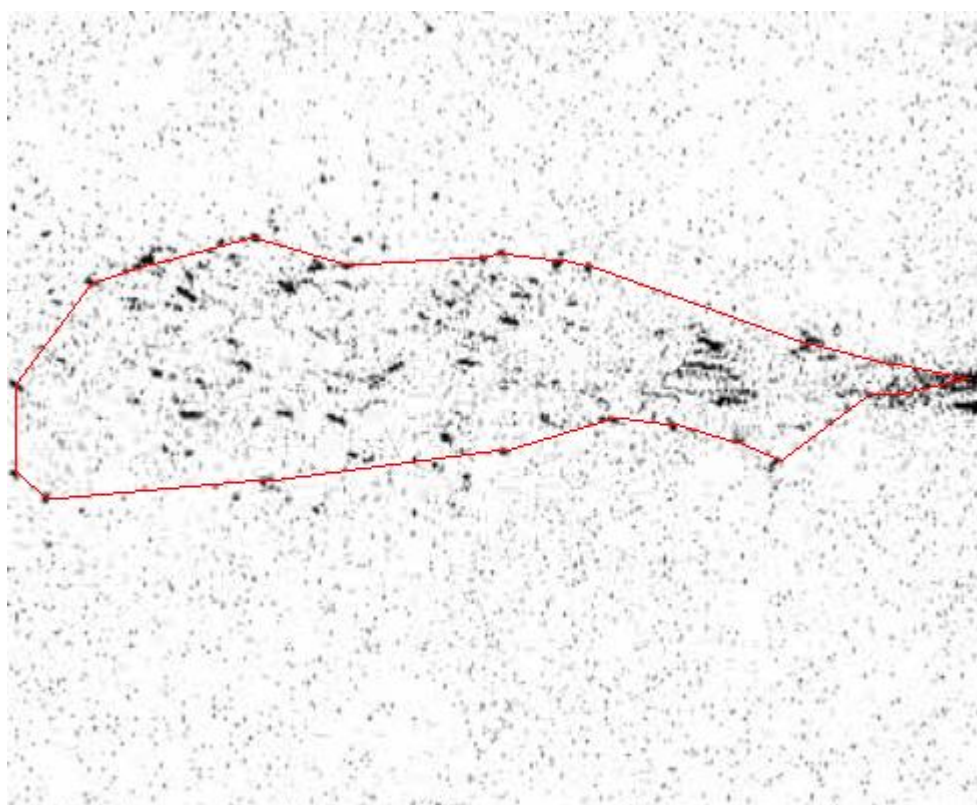
V následující tabulce 8 lze vidět souhrnné výsledky celého algoritmu. Pro každou variantu bylo provedeno 50 různých pokusů. Atribut „úspěšnost“ je subjektivní ohodnocení člověkem, zda je dosažená hranice dostatečně věrohodná. Hranice nemusí odpovídat přesně hranici, kterou by sestrojil člověk, ale musí proud reprezentovat tak, aby byla použitelná pro další vyhodnocení. Atribut „Průměrný počet generačních obměn“ je daný počtem generačních obměn, po kterých byl vyselektován jedinec, kterého se již nepodařilo překonat. Hodnota odpovídá celkovému počtu generací, ale je snížena o nadefinovaný počet populačních obměn, kdy algoritmus nejlepšího jedince již nepřekonal (50 či 100). Při populační obměně byla vždy nahrazena polovina populace, tedy 100 z 200 jedinců.

Tabulka 8 Výsledky testování genetického algoritmu

Počet bodů (= počet genů chromozomu)	Počet možných variant hranice	Průměrný počet generačních obměn (zaokrouhleno)	Úspěšnost [%]
20	1 048 576	41	96
40	1 099 511 627 776	175	82
60	1 152 921 504 606 846 976	442	76
120	$1,33 \times 10^{36}$	3729	58

Pro větší počty bodů je počet všech řešení tak velký, že je vhodné dát algoritmu více příležitostí ke zlepšení. 50 generací bez zlepšení nejlepšího řešení nastavených pro 20 a 40 bodů je z tohoto pohledu nedostatečný počet. Proto byla pro 60 a 120 bodů ukončovací podmínka zvětšena na 100 či 200 generací beze změny nejlepšího. Zvýšení této hodnoty vede k vyšší úspěšnosti algoritmu, ale samozřejmě též k nárůstu doby optimalizace.

Výsledek genetického algoritmu pro reálný snímek 33a je na snímku 53.



Snímek 53 Výsledek genetického algoritmu pro reálný snímek

7 URČENÍ RYCHLOSTI PROUDU VIZUALIZOVANÉHO MLHOU

Pro účel určení rychlosti proudu ve snímcích prvního typu existuje postup, využívající vzájemnou korelaci. Lze jej použít jak na případ vizualizace aditivy, která tvoří souvislá vlákna (mlha, kouř...), tak na případ vizualizace látkami, které souvislá vlákna netvoří (heliové bubliny, saze, jiskry...). V druhém případě je však taková evaluace závislá na množství vizualizačních částic proudu. Pro ne příliš husté proudy byla vyvinuta jiná možnost, o níž bude pojednávat kapitola 8. Naopak pro snímky proudění vizualizované mlhou je použití vzájemné korelace (či její obdoby) výhodným způsobem řešení.

Tato kapitola využívá poznatky Bracewella (1965) a Kapincheva, Bradua, Barnese et al. (2015). Nastíní základní princip použití korelace na snímek proudu vizualizovaný mlhou.

7.1 PRINCIP POUŽITÍ VZÁJEMNÉ KORELACE

Mlha, jíž se vizualizuje proudící vzduch, není homogenní, ale obsahuje různé barevné odstíny v různých stupních šedi. To může být způsobeno jednak nečistotou mlhy, jednak její nekonzistentní hustotou. Sledováním změny polohy nuancí mlhy je možné vyvozovat rychlost proudění v konkrétním místě.

Pro určení rychlosti v konkrétním bodě Q je třeba sledovat jeho blízké okolí, typicky čtvercové s hranou o $2h + 1$ pixelech, kde h představuje největší rozdíl souřadnice (X či Y) oproti souřadnicím vyšetřovaného bodu. Okolí vyšetřovaného bodu by nemělo být příliš malé, aby jej bylo možné identifikovat v následujícím snímku. Obecně však rychlost pixelů v takovém okolí není zcela stejná, což je negativně ovlivněno právě rostoucí velikostí oblasti.

Vektor změny polohy je určován ve dvojici po sobě jdoucích snímcích S^0 a S^1 . Díky známé časové diferenci je poté z tohoto vektoru vypočítána konkrétní rychlost. V prvním snímku S^0 je kolem bodu Q , v němž má být detekována rychlost, stanovena oblast $(2h + 1) \times (2h + 1)$. V následujícím snímku S^1 je pak tato oblast vyhledávána. Výchozím bodem jsou při tom opět souřadnice vyšetřovaného bodu Q . Ve snímku S^1 je kolem zmíněných souřadnic vyhledávána co nejpodobnější oblast oblasti ve snímku S^0 . Existuje více možností, jak realizovat samotné porovnání vyšetřované oblasti ve snímku S^0 a uvažované oblasti ve snímku S^1 . Některé popisuje například Bracewell (1965) a Kapinchev, Bradu, Barnes, et al. (2015). V této práci probíhá toto porovnání násobením jasu pixelů o stejných relativních souřadnicích a sčítáním těchto součinů. Tím je získáno ohodnocení oblasti *EVAL*. Oblast, která má ze všech oblastí největší hodnotu tohoto ohodnocení, je prohlášena za následovníka vyšetřované oblasti. Střed oblasti ve snímku S^1 je pak druhou polohou vyšetřovaného bodu Q . Prohledávané okolí definuje kladná celočíselná konstanta m , která představuje největší možný rozdíl souřadnice (X či Y) mezi vyšetřovaným bodem Q a uvažovaným bodem P . Hodnota m by měla být volena dostatečně vzhledem k rychlosti proudu. Negativně však ovlivňuje procesní čas. Také zvyšuje množství uvažovaných bodů, z nichž je vybírána druhá poloha vyšetřovaného bodu. Vybrány pak mohou být i příliš vzdálené body, jejichž okolí je však podobné okolí bodu Q .

Algoritmus pro konkrétní bod $Q = [Q_x, Q_y]$ lze zapsat následovně (syntaxe Delphi):

```
//procházení uvažovaných bodů
FOR Xm := QX-m TO QX+m DO
  FOR Ym := QY-m TO QY+m DO
    BEGIN
      EVAL := 0; //ohodnocení podobnosti obou okolí
      //procházení relativních souřadnic okolí
      FOR Xh := -h TO h DO
        FOR Yh := -h TO h DO
          BEGIN
            IQ := getPixelIntensity(QX+Xh, QY+Yh);
            Ih := getPixelIntensity(Xm+Xh, Ym+Yh);
            EVAL := EVAL + IQ * Ih; //(*)
          END;
        IF EVAL > EVALmax THEN
          BEGIN
            EVALmax := EVAL;
            PX := Xm;
            PY := Ym;
          END;
        END;
      END;
```

Algoritmus výše je vhodné vylepšit tím, že se jako první uvažovaný následovník bodu Q bude uvažovat bod ve snímku S^1 se souřadnicemi $[Q_x, Q_y]$. Dále bude následovat procházení pole $(2m+1) \times (2m+1)$, v němž se bod $[Q_x, Q_y]$ již přeskočí. Tím se ošetří rychlosti v okolí proudu, kde se nenachází žádná mlha a pozadí je zde homogenní. Pro všechny uvažované body tak vychází stejná hodnota $EVAL$. $EVAL$ hodnota první vyšetřovaného bodu už nikdy nebude překonána a bez tohoto opatření by jím byl bod $[Q_x - m, Q_y - m]$. Detekovaná změna polohy by v bodě $[Q_x, Q_y]$ tedy byla $[-m, -m]$.

Další vylepšení je možné při napočítávání hodnoty $EVAL$ nejen ze součinu intenzit odpovídajících pixelů, ale i ze součinů jejich převrácených hodnot (intenzit negativní barvy). Řádek komentovaný v kódu (*) je tedy vhodné nahradit tímto:

```
EVAL := EVAL + IQ * Ih + (255 - IQ) * (255 - Ih); //(**)
```

Chování evaluační funkce s původním výrazem (*) je patrné na následujícím příkladu:

Oblast vyšetřovaného bodu

$$O^0_Q = \begin{bmatrix} 255 & 255 & 255 \\ 0 & 255 & 255 \\ 255 & 0 & 255 \end{bmatrix}$$

Uvažované oblasti

$$O^1_{P_0} = \begin{bmatrix} 255 & 255 & 255 \\ 255 & 0 & 255 \\ 255 & 255 & 0 \end{bmatrix}$$

$$O^1_{P_1} = \begin{bmatrix} 255 & 255 & 255 \\ 0 & 255 & 255 \\ 255 & 0 & 255 \end{bmatrix}$$

$$O^1_{P_2} = \begin{bmatrix} 255 & 255 & 255 \\ 255 & 255 & 255 \\ 255 & 255 & 255 \end{bmatrix}$$

Při výpočtu evaluační hodnoty po oblast $O^1_{P_0}$ vyjdou čtyři nulové součiny $IQ \times Ih$, a to pro tyto relativní souřadnice:

- $[-1, 0] = 0 \times 255$;
- $[0, -1] = 0 \times 255$;
- $[0, 0] = 255 \times 0$;
- $[-1, -1] = 255 \times 0$;

Zato pro oblast $O^1_{P_1}$ jsou nulové součiny pouze 2:

- $[-1, 0] = 0 \times 0$;
- $[0, -1] = 0 \times 0$;

V porovnání oblastí $O^1_{P_0}$ a $O^1_{P_1}$ lze tedy vidět, že je výhodné, když mají pixely se shodnou intenzitou stejné relativní souřadnice. Koncepti ale rozbíjí třetí oblast $O^1_{P_2}$ se všemi pixely pouze o nejvyšší možné intenzitě, čímž je zaručen nejvyšší možný součin (*). Přestože oblast $O^1_{P_1}$ je identická se zdrojovou oblastí, oblast $O^1_{P_2}$ se jí velikostí evaluační funkce vyrovná, když pro ni vyjdou také pouze dva nulové součiny:

- $[-1, 0] = 0 \times 255$;
- $[0, -1] = 0 \times 255$;

Z toho důvodu je důležité připočítávat i součin inverzních intenzit (**). Pro oblast $O^1_{P_1}$ vychází dva nenulové součiny inverzních souřadnic:

- $[-1, 0] = 255 \times 255$;
- $[0, -1] = 255 \times 255$;

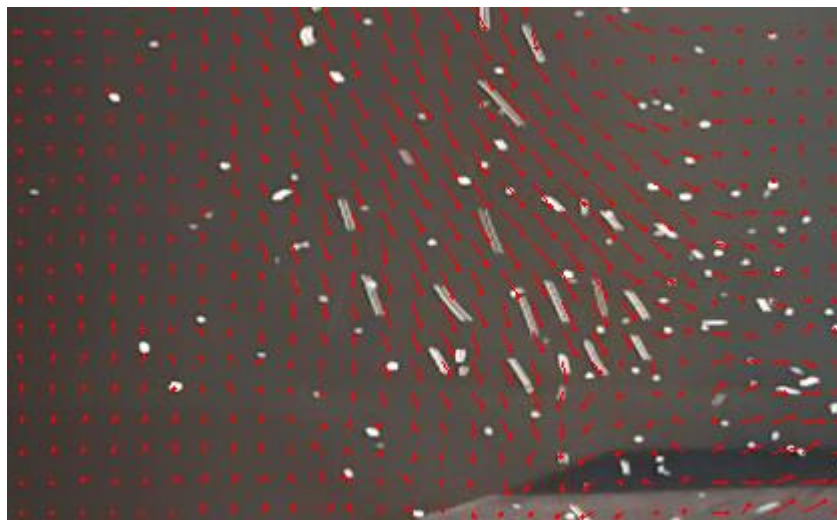
Zatímco pro oblast $O^1_{P_2}$ jsou všechny součiny inverzních intenzit nulové.

7.2 VEKTOROVÁ MAPA

V předchozí kapitole byl pro konkrétní bod $Q = [Q_X, Q_Y]$ ve snímku S^0 bod jeho následující polohy ve snímku S^1 v okolí bodu $[Q_X, Q_Y]$ definován konstantou m . Výběr tohoto bodu byl prováděn na základě porovnání oblasti kolem bodu Q ve snímku S^0 (definované konstantou h) s oblastí kolem každého uvažovaného bodu ve snímku S^1 . Tyto poznatky lze využít pro sestavení tzv. **vektorové mapy**. Jedná se o síť bodů, vzdálených od sebe o konkrétní konstantní vzdálenost g ,

pro které je určena rychlost využitím vzájemné korelace. Platí, že vektor rychlosti lze určit v každém bodě snímku (Keating, Wolf, Scarpace, 1975).

Výsledek vektorové mapy je patrný například na snímku 54, kde byla metoda aplikována na proudění heliových bublin.



Snímek 54 Příklad vektorové mapy

7.3 DALŠÍ MOŽNOSTI URČENÍ RYCHLOSTI

Použití vzájemné korelace je jen jednou z možných metod, jak řešit určení vektorů rychlosti. Existuje například více různých typů korelace, z nichž každá následuje obdobný princip, jako byl popsán v kapitole 7.1.

Pokud lze v proudu detekovat částice, lze také v určité oblasti určit všechny možné vektory rychlosti, tedy ke každé částici v oblasti ve snímku S^0 určit vektory do všech částic v této oblasti ve snímku S^1 . Z množiny vektorů je pak vybrán nejčtenější. Tento způsob využívá například Pavelek, Janotková a Štětina (2001).

8 URČENÍ RYCHLOSTI PROUDU VIZUALIZOVANÉHO HELIOVÝMI BUBLINAMI

Protože metoda použití vzájemné korelace je založena na sledování nuancí mlhy v po sobě jdoucích snímcích, lze ji principiálně použít i na záznam proudění vizualizovaného heliovými bublinami. Úspěšnost algoritmu vzájemné korelace se odvíjí od hustoty proudu heliových bublin. Každá částice představuje použitelnou informaci, naopak pozadí o proudu příliš nevypovídá. Pokud je hledaná matice snímku S^0 různorodá (díky množství bublin), je ve snímku S^1 identifikovatelná lépe než například matice okolí s jedinou bublinou. Otázkou tedy je, jak určit směr pohybu bublin i ve snímcích s menší hustotou bublin. Dalším úskalím korelačních metod je také jejich práce s 2D objekty, kdy se v místech o téměř stejných souřadnicích X a Y vyskytují bubliny s různými rychlostmi. To je způsobeno jejich různou polohou ve směru osy Z , což však ve 2D snímku není dostatečně patrné.

8.1 SLEDOVÁNÍ ČÁSTIC

Pokud se směr proudění pokouší určit člověk, podvědomě dokáže řešit problém vzájemnou korelací. Pokud je však bublin obecně málo, přechází řešení na sledování jednotlivých bublin, jehož princip lze zalgoritmovat pro počítačové vyhodnocení. Takovou metodu představují Šťastný a Richter (2014).

Ke správnému fungování algoritmu pro základní snímek S^0 je zapotřebí ještě dvou dalších po něm následujících snímků S^1 a S^2 . Snímek S^0 bude generátorem bublin. Snímek S^1 bude generátorem cest. Snímek S^2 bude hodnotitelem cest. Mezi snímky musí být stejná velmi malá časová difference dt .

8.1.1 Inicializace procesu

Jak již bylo řečeno, do procesu sledování částic vstupují celkem tři snímky. Ve všech třech však bude na bubliny nahlíženo jako na body reprezentované reprezentanty, jako tomu bylo v kapitole 6. Ve všech třech snímcích jsou provedeny kroky „Určení pixelů náležících heliovým bublinám“ (kapitola 6.1.1) a „Výběr reprezentantů heliových bublin“ (kapitola 6.1.2). Na tomto místě je nutno zdůraznit, že algoritmus vyžaduje reprezentanty bublin, nikoliv reprezentanty shluků bublin. Krok „Separace neplatných reprezentantů“ (kapitola 6.1.3) již není nutné vykonávat. Že je dotyčná bublina neplatná, bude zřejmé díky dalším snímkům, v nichž bude její poloha neměnná. Rychlost reprezentantů, jejichž poloha se nezmění, bude nulová. Není tedy důležité, zda reprezentují bublinu mimo proud či dokonce parazitní nehybný objekt. Řešení by mohly degradovat pouze pohybující se parazitní objekty.

Výsledkem inicializační fáze jsou tak tři množiny reprezentantů M^0 , M^1 , M^2 . Množina bublin M^0 ve snímku S^0 definuje, jaké bubliny budou v následujícím procesu sledovány. Protože však může nastat situace, kdy bublina ve snímku S^1 či S^2 opustí hranice snímku a není tedy možné ji sledovat, existuje možnost ze sledování vyloučit bubliny na okrajích snímku, protože by jim v procesu mohl být přisouzen nesprávný směr. Pokud je sekvence pořízena z dostatečné perspektivy a s dostatečně malou časovou diferencí mezi snímky, riziko, že vyloučením bublin poblíž hranice snímku bude vyloučena i bublina vyšetřovaného proudu je minimální.

8.1.2 Generování možných cest

V předchozím kroku byla ve snímku S^0 určena množina reprezentantů bublin M^0 , jejichž pohyb má být sledován. Snímek S^1 bude definovat možné cesty, kterými se tyto bubliny mohly vydat.

Je dán reprezentant R_i^0 množiny M^0 o souřadnicích $[X_i^0, Y_i^0]$. Základní otázkou je, kam se jím reprezentovaná bublina přemístila. Je tedy nutné najít její polohu ve snímku S^1 . Nová poloha reprezentanta R_i^0 je hledána v množině reprezentantů M^1 jako poloha jednoho z reprezentantů R_j^1 . Správný reprezentant R_j^1 je očekáván v okolí polohy bodu R_i^0 tedy v blízkosti $[X_i^0, Y_i^0]$. Proto je vytipováno u reprezentantů množiny M^1 , které jsou souřadnici $[X_i^0, Y_i^0]$ nejbližší. Počet u je nutné zvolit tak, aby bylo jisté, že se mezi vybranými nachází správný reprezentant – reprezentant téže bubliny ve snímku S^1 . Hodnotu u ovlivňují tyto faktory:

- **Hustota bublin v proudu** – Pokud je proud bublin hustý co do jejich počtu, bude v okolí polohy bodu R_i^0 velké množství bublin. Proto je třeba uvažovat větší množství potenciálních bublin, aby bylo zaručeno, že mezi nimi bude ta správná.
- **Rychlost proudění ve ventilačním systému** – Touto rychlostí je myšlena rychlost, se kterou proud opouští ventilační systém. Z té lze usuzovat na rychlost, s jakou heliové bubliny mění svoji polohu po opuštění ventilačního systému. S vyšší možnou rychlostí bubliny se zvětšuje oblast, v níž lze předpokládat tuto bublinu v následujícím snímku. Větší uvažovaná oblast pak bude obsahovat větší počet bublin. Faktor rychlosti proudění tedy zvyšuje počet bublin u ve snímku S^1 , které by mohly být bublinou R_i^0 snímku S^0 .
- **Časová difference mezi snímky** – Tento faktor má přesně opačný efekt oproti faktoru rychlosti proudění, protože zmenšuje oblast ve snímku S^1 , kam se mohla přesunout konkrétní bublina.

Obecně lze číslo u zvolit pro husté proudy 10, pro méně husté 5. Pokud by bylo možné dle rychlosti proudění z vyústky a dle časové difference stanovit maximální vzdálenost d_{imax} , kam nejdál se může bublina přesunout, místo daného počtu u by bylo možné vybrat všechny reprezentanty R_j^1 , nacházející se ve snímku S^1 v kruhové oblasti se středem v bodě $[X_i^0, Y_i^0]$ a poloměrem d_{imax} . Rychlost proudění z vyústky je však větší než maximální rychlost proudění v bodě $[X_i^0, Y_i^0]$, kterou je obtížné odhadnout. I tak však může být vzdálenost d_{imax} parametrem procesu namísto parametru k . V této práci je přesto používán parametr k , a to většinou o hodnotě 5, pro husté proudy případně 10.

V tomto kroku tedy bylo každému reprezentantovi R_i^0 množiny M^0 přisouzeno u nejbližších reprezentantů R_j^1 množiny M^1 .

8.1.3 Rozvíjení cesty

Pro každého reprezentanta R_i^0 snímku S^0 je k dispozici u pozic, v nichž by se mohl nacházet ve snímku S^1 . Existuje tedy u cest, kterými se mohla bublina vydat. Každou takovou cestu je nutno prověřit, zda je cestou, kterou bublina skutečně putuje. Pokud se bublina dostala z bodu $[X_i^0, Y_i^0]$ daného reprezentantem R_i^0 do bodu $[X_j^1, Y_j^1]$ daného reprezentantem R_j^1 , ideálně by tak tutéž změnu polohy měla vykonat během mezi snímek S^1 a S^2 . Ve snímku S^2 bude tedy očekáván reprezentant v blízkosti bodu R_j^1 $[X_j^1, Y_j^1]$, kde

$$X_j^{1'} = X_{1j} + (X_{1j} - X_{0i}) = 2 \cdot X_{1j} - X_{0i} \quad (8.1)$$

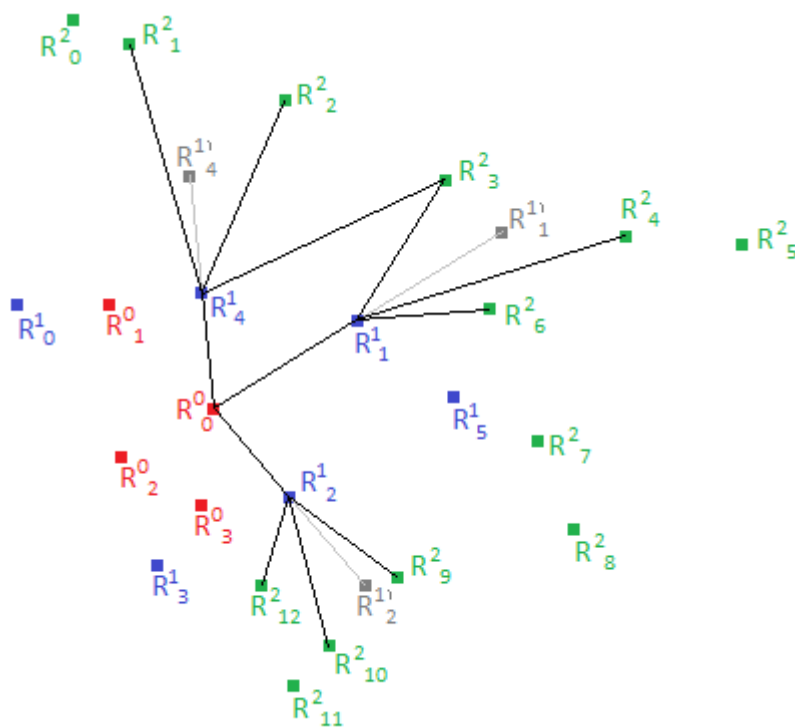
$$Y^{1'}_j = Y_{1j} + (Y_{1j} - Y_{0i}) = 2 \cdot Y_{1j} - Y_{0i} \quad (8.2)$$

Je velmi nepravděpodobné, aby změna polohy bubliny mezi snímky S^0 a S^1 byla naprosto shodná se změnou polohy téže bubliny mezi snímky S^1 a S^2 . Proto je třeba prohledat blízké okolí bodu $R^{1'}_j$ a hledat v něm reprezentanty z množiny M^2 . V tomto případě je třeba vykonat podobný úkol jako v kapitole 8.1.2, tedy zvolit počet reprezentantů v , které by mohli odpovídat poloze reprezentanta R^0_i ve snímku S^2 . Hodnota v bude závislá především na hustotě heliových bublin v proudu, protože bod $R^{1'}_j$ je určen s ohledem na předchozí změnu polohy mezi snímky S^0 a S^1 . Nicméně za vzorkovací čas dt může dojít ke změně směru i rychlosti proudění, což zvětšuje množinu možných poloh bubliny ve snímku S^2 . Opět tedy lze jako parametr použít maximální poloměr d_{ijmax} , definující oblast ve snímku S^2 , do níž by se mohl přesunout reprezentant R^0_i , který se předtím přesunul do bodu $R^{1'}_j$. Typicky se pro vyhledávání blízkých bodů používá stejné nastavení jako v kapitole 8.1.2, tedy maximální poloměr oblasti $d_{ijmax} = d_{imax}$. V práci je však pro obě vyhledávání použitý konkrétní počet nejbližších bodů $v = u = 5$ (případně 10).

V tomto bodě tedy bylo ke každé z u cest určeno v možných pokračování cest. Schematicky to znázorňuje příklad na snímku 55, kdy jsou v jednom obrázku vyobrazeni reprezentanti snímku S^0 – červeně, S^1 – modře a S^2 – zeleně. Dále jsou zde zobrazeny varianty cest z bodu R^0_0 pro $u = v = 3$. Nejbližší body z množiny M^1 bodu R^0_0 jsou R^{1_1} , R^{1_2} a R^{1_4} . Například pro bod R^{1_1} postup pokračuje nalezením bodu $R^{1'1_1}$, který na polopřímce $R^0_0 R^{1_1}$ ve dvojnásobné vzdálenosti od bodu R^0_0 , než je bod R^{1_1} . Očekáváme, že pokud se bublina za čas dt přesunula z polohy dané R^0_0 do polohy dané R^{1_1} , pak by se z pozice R^{1_1} měla za stejný čas dt přesunout do bodu $R^{1'1_1}$. Neblíže bodu $R^{1'1_1}$ jsou body R^{2_3} , R^{2_4} a R^{2_6} . Podobným způsobem jsou rozvinuty i cesty pro body R^{1_2} a R^{1_4} .

Poznámka: Bod R^{2_3} figuruje jak v cestě R^0_0 - R^{1_1} - R^{2_3} , tak v cestě R^0_0 - R^{1_4} - R^{2_3} , což je však povoleno.

Princip generování cest užitím snímku S^1 (kapitola 8.1.2) a rozvíjení cest užitím snímku S^3 je tedy podobný, a sice v hledání určitého počtu nejbližších bodů konkrétnímu bodu. Ale zatímco v prvním případě je tímto bodem přímo bod cesty (vyšetřovaný bod), ve druhém případě je bodem uměle vytvořený bod s předpokladem, že se do něj vyšetřovaný bod dále přesune.



Snímek 55 Ilustrace generování cest pro bod R^0_0 snímku S^0 pro $u = v = 3$.

8.1.4 Ohodnocení cest

Výstupem předchozího kroku je $u \times v$ cest pro každého reprezentanta množiny M^0 . Pokud byly konstanty u a v voleny adekvátně a pokud reprezentovaná bublina neopouští hranice snímku, právě jedna z těchto cest je platná. Každou cestu je proto nutné ohodnotit, přičemž nejkvalitnější cesta bude prohlášena za platnou. Pokud bude platit cesta $R^0_i R^1_j R^2_k$, pak hledaným vektorem rychlosti v bodě R^0_i je $R^0_i R^1_j$. Pokud je časová diference dt malá, pak lze očekávat, že by změna $R^0_i R^1_j$ a $R^1_j R^2_k$ měly být velmi podobné (ostatně je to předpokladem celé metody). Proto je cesta ohodnocena následujícími kritérii:

- **Změna rychlosti** – Kritérium vychází z předpokladu, že bublina by neměla výrazně zpomalit ani zrychlit. Faktor vzdálenosti získáme jako poměrnou změnu délky (absolutní) vztaženou k délce první části cesty:

$$f_v = \frac{||R^0_i R^1_j| - |R^1_j R^2_k||}{|R^0_i R^1_j|} \quad (8.3)$$

Přičemž ideální hodnota faktoru vzdálenosti je 0.

- **Změna směru** – Toto kritérium hodnotí změnu směru letu bubliny mezi snímky S^1 a S^2 oproti původnímu směru danému S^0 a S^1 . Hodnota faktoru f_ϕ je rovna velikosti úhlu, které svírají vektory $R^0_i R^1_j$ a $R^1_j R^2_k$. Úhel bude vždy v rozmezí $<0, \pi>$, přičemž jeho ideální hodnotou je 0.
- **Změna velikost bubliny** – Každý z trojice reprezentantů cesty se odkazuje na konkrétní bublinu. Reprezentovanou bublinou se rozumí množina všech bodů vyhodnocených jako body bubliny, které mají k danému reprezentantovi ze všech reprezentantů nejbližší (více

se tomuto bodu věnuje kapitola 6.1.2). Faktor změny velikosti bubliny f_c je pak vypočítán následovně:

$$f_c = \frac{|C_i^0 - C_j^1| + |C_i^0 - C_k^2|}{C_i^0} \quad (8.4)$$

Kde C_i^0 – počet pixelů i -té bubliny ve snímku S^0

C_j^1 – počet pixelů j -té bubliny ve snímku S^1

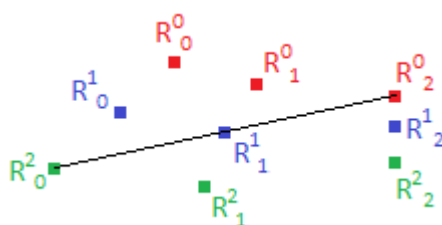
C_k^2 – počet pixelů k -té bubliny ve snímku S^2

Ideální hodnota faktoru f_c je 0.

Faktor změny velikosti bubliny však nelze doporučit všeobecně, protože správnost algoritmu výběru reprezentantů nemusí být stoprocentní. Například jediná bublina ve snímku S^0 může být algoritmem kapitoly 6.1.2 ve snímku S^1 rozdělena na dvě. Reprezentanti těchto dvou bublin budou velmi blízko u sebe, a proto jsou i cesty k nim či od nich prakticky totožné. Faktory změny rychlosti i změny úhlu budou pro obě cesty téměř shodné a nebude záležet, která z cest bude vybrána jako výhodnější. Faktor změny velikosti bubliny však bude vypadat diametrálně jinak a může hodnotu cest devalvovat.

- **Vzdálenost následovníka** – I samotná délka první části cesty může být kritériem. Čím menší je vzdálenost mezi body $R^0_i R^1_j$, tím je pravděpodobnější, že bod R^1_j je skutečným následovníkem bodu R^0_i . Pokud je díky koeficientům u a v vyšetřováno velké množství cest, mohou být uvažovány i vzdálení nepravděpodobní následovníci. Nejbližší body by proto měly být zvýhodněny, aby nedošlo k případům, jako zachycuje snímek 56. Při hledání cesty z bodu R^0_2 je nalezena cesta $R^0_2 R^1_1 R^2_0$, která má obě úsečky cesty stejně dlouhé a stejně orientované. Pokud hodnoticí funkce nebude zohledňovat vzdálenost následovníka, bude mít zmíněná cesta nejlepší možné ohodnocení 0. Oproti tomu cesta $R^0_2 R^1_2 R^2_2$ (správná) je sice také velmi kvalitní, avšak úsečka $R^1_2 R^2_2$ je mírně delší než $R^0_2 R^1_2$. Proto tato cesta bude mít ohodnocení větší než 0, a tedy nebude vybrána. Obecně lze říci, že se zvyšujícím se množstvím heliových bublin v proudu a se zvyšujícími se hodnotami konstant u a v roste pravděpodobnost, že bude nalezena nesprávná cesta s velmi nízkým ohodnocením.

Faktor vzdálenosti následovníka tak zároveň plní funkci ochrany proti nevhodně zvoleným konstantám u a v pro konkrétní bublinu. Faktor vzdálenosti následovníka f_d je tedy roven vzdálenosti $R^0_i R^1_j$, přičemž jeho ideální hodnota je 0.



Snímek 56 Ideální cesta, chybně nalezená z důvodu chybějícího faktoru vzdálenosti následovníka

Celkové ohodnocení cesty f_p pak spočívá v sečtení zmíněných faktorů vynásobených příslušnými váhovými koeficienty.

$$f_p = k_v \cdot f_v + k_\varphi \cdot f_\varphi + k_C \cdot f_C + k_d \cdot f_d \quad (8.5)$$

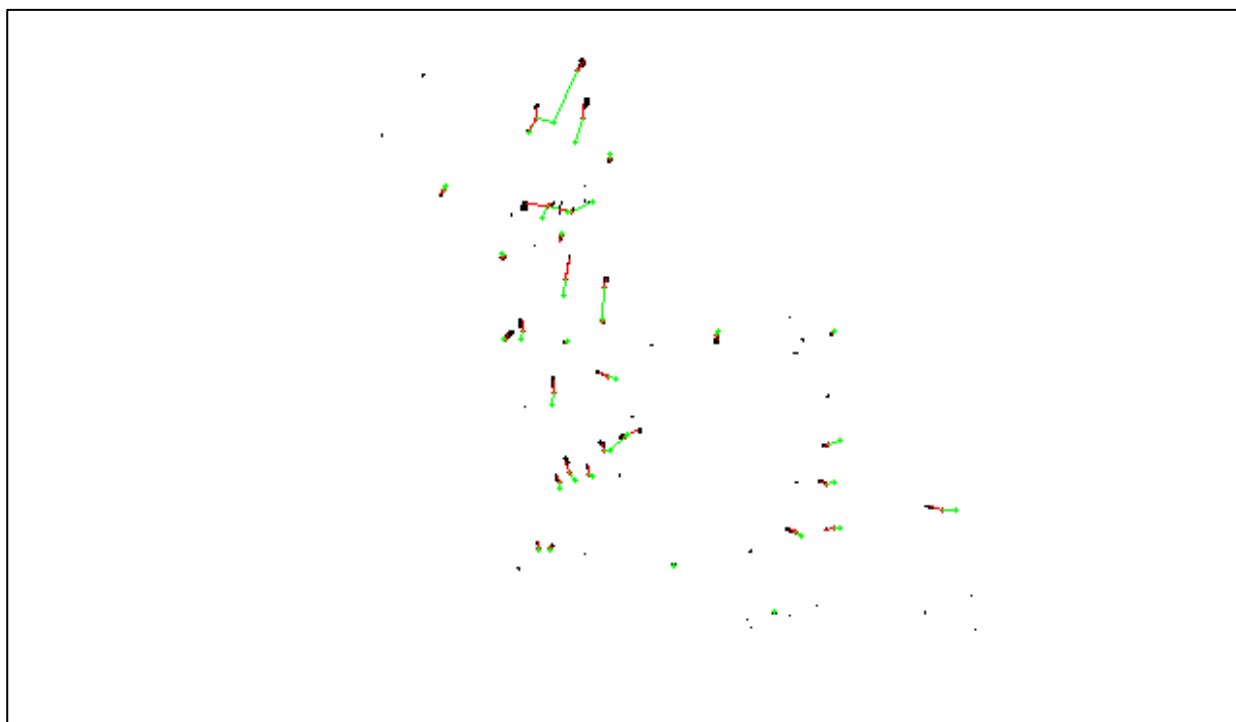
Konkrétní hodnoty váhových koeficientů je třeba volit dle konkrétních popsaných vlastností proudění (rychlost proudění, hustota bublin v proudě, časová difference mezi snímky dt). Obecně by velký vliv na výsledek měl mít koeficient vzdálenosti následovníka f_d , avšak protože je jeho hodnota uvedena absolutně a nikoliv poměrně, musí být hodnota k_d malá. Koeficienty změny směru a změny rychlosti je mu pak nutno přizpůsobit. Otázkou je také srovnání, jaká změna úhlu je co do kvality odpovídající jaké změně rychlosti. Faktor změny velikosti bubliny může být diskutabilní kvůli zmíněné chybovosti, avšak lze jej použít jako doplňujícího hodnotitele cesty.

Pro snímek 57 byly použity tyto váhové koeficienty: $k_v = 0.5$; $k_\varphi = 1$; $k_C = 0.1$; $k_d = 0.2$

Cestu lze hodnotit i dalšími faktory, které však nelze doporučit. Sledování tvaru bubliny by bylo nesmírně náročné na strojový čas. Navíc se tento tvar může velmi rychle změnit, a tak je přínos takového faktoru diskutabilní. Z důvodu nestálého tvaru bubliny nemá význam řešit ani délku a šířku bubliny, k jejichž důslednému stanovení by bylo nutné řídit se také orientací bubliny. Místo tvaru či rozměrů bubliny se pro její popis lépe hodí právě počet jejích pixelů, více vypovídající o její mohutnosti, která zůstává neměnná (jedná se o tutéž bublinu jen v jiných snímcích).

Jak již bylo řečeno, celý proces v této kapitole slouží k určení správných vektorů rychlosti bublin ve snímku S^0 při posunu do poloh ve snímku S^1 s využitím snímku S^2 . Pro vyhodnocení rychlostí ve snímku S^1 by bylo nutno použít jemu následující snímky S^2 a S^3 .

Výsledek celého procesu hledání vektorů rychlostí ve snímku S^0 pomocí snímků S^1 a S^2 lze vidět na snímku 57. Detekované vektory rychlostí jsou znázorněné červeně. Zeleně jsou pak pro úplnost znázorněny následující úsečky v dotýčných cestách (mezi polohami bublin ve snímku S^1 a S^2). Ty však mají pouze pomocný charakter a nejsou výstupem algoritmu, neboť vektory rychlostí pro snímek S^1 budou stanoveny v novém procesu s využitím snímků S^2 a S^3 .



Snímek 57 Výsledek detekce vektorů rychlosti, kde:

červeně jsou označeny detekované vektory, zeleně pokračování v cestě

I ze snímku 57 si lze povšimnout, že ne všechny vektory byly určeny správně. Po validaci člověkem byly odhaleny tři chybné vektory. Při testování metody na dvaceti snímcích dvou různých sekvencí byla úspěšnost metody průměrně 92 % (tedy 92 % bublin mělo očekávaný správný vektor).

8.2 ZPRACOVÁNÍ DETEKOVANÝCH VEKTORŮ

Jak bylo řečeno, metoda sledování částic není stoprocentně přesná. Z uvedeného testování plyne, že cca 8 % bublin má chybně určenou rychlost. Výsledky jsou i tak použitelné pro další zpracování. Lze však provést mezikrok a jejich vhodnou interpretací chyby korigovat.

8.2.1 Porovnávání okolních vektorů

Ve 2D snímku proudění heliových bublin mohou existovat parazitní bubliny s velmi nízkou rychlostí, které jsou sice mimo proud, ale zůstávají na pozadí snímku. Pokud se souřadnice X , či Y takové bubliny výrazně liší od bublin v proudu, pak tato bublina popis rychlosti neovlivní, neboť díky ní potvrdíme, že je v dotyčném místě rychlost proudění nulová. Pokud by však souřadnicemi X a Y spadala do oblasti proudu, její polohu mimo proud by odhalila až souřadnice Z , která však ve 2D snímku není k dispozici.

Uvedený důvod spolu z chybovostí vede k možnosti porovnávat mezi sebou vektory blízkých bublin. Pokud má jedna z bublin výrazně vyšší nebo výrazně nižší rychlost než všechny ostatní (neexistuje žádná, která by měla podobnou rychlost), pak vektor rychlosti označíme za chybný. Správná hodnota vektoru rychlosti této bubliny už v této fázi není znovu odvozována. Postačí, když bublina a její vektor budou z množiny výsledků vyloučeny, aby je nezkreslovaly. Rychlost v tomto místě tedy nebude chybně určena, jen bude ponechána neznámá.

Pro tento účel je nutné definovat poloměr uvažovaného okolí a parametry porovnání vektorů. Lze porovnávat obě souřadnice vektoru, nebo směr a velikost, nebo například pouze velikost. Je také nutné stanovit minimální počet bublin v okolí. Pokud jsou v okolí například jen dva vektory o různých velikostech, pak nelze rozhodnout, který z nich by se měl smazat.

8.2.2 Mazání kolizních vektorů

Jedná se o elementární hledání chyby, kdy jsou porovnávány dva různé vektory dvou různých bublin. Pokud je dle polohy reprezentantů a změny polohy vypočítáno, že bubliny se budou v následujícím snímku nacházet v téže pozici, došlo by ke kolizi. Tento stav svědčí o tom, že pro obě různé bubliny byl vybrán stejný následovník, tedy alespoň v jednom případě byl vybrán chybně. Zpracování tohoto stavu bude nastíněno pro dva kolizní vektory, přestože obecně mohou v jenom místě kolidovat dokonce všechny vektory:

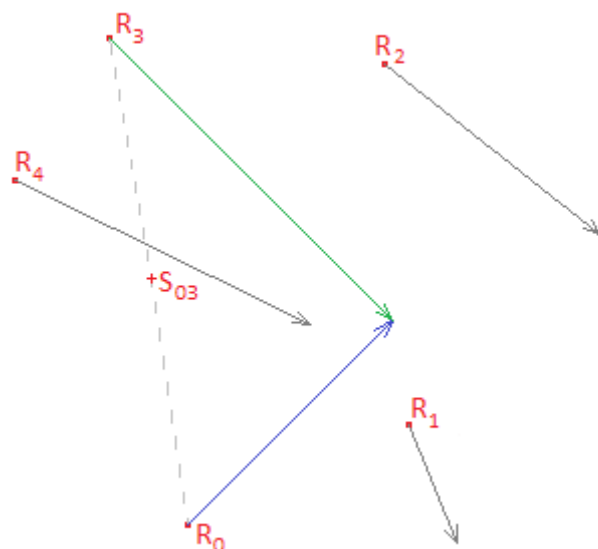
1. **Vyloučení obou bublin a jejich vektorů ze zpracování** – Jedná se o nejjednodušší způsob řešení. Pokud nelze rozhodnout, který vektor je špatně určený a který správně, vyloučení budou oba, což představuje jistotu, že další zpracování výsledků nebude ovlivněno chybou. Výhodou je i rychlost ošetření tohoto problému. Negativem je, že smazáním budou ztraceny dva vektory ze stejné oblasti, což může být významná ztráta informace.
2. **Přisouzení průměru vektorů oběma reprezentantům** – Tuto možnost lze použít, pokud jsou oba vektory podobné směrem i velikostí. V tom případě mohly vzniknout tak, že ve snímku S^0 je bublina rozdělena na dva reprezentanty, zatímco ve snímku S^1 v ní byl nalezen pouze jediný reprezentant. V takových případech lze buď oba vektory ponechat, nebo je zprůměrovat. Průměrování vektorů lze použít i v případě odlišných vektorů, což bude znamenat korekci chyby správným vektorem.
3. **Vymazání chybného vektoru** – V tomto případě je rozhodováno na základě okolí, který z vektorů je správný a který nikoliv. Okolím se bude rozumět n (například 3) reprezentantů, kteří jsou nejbližší středu mezi reprezentanty kolizních vektorů. V okolí nebudou uvažovány oba kolizní vektory. Pro vektory okolních bublin bude stanoven průměrný vektor. Ten z kolizních vektorů, který bude mít menší rozdíl souřadnic dS od průměru bude ponechán, druhý vektor nebude uvažován. Hodnota dS se pro oba kolizní vektory stanoví takto:

$$dS = \left| X - \frac{\sum_{i=0}^{n-1} X_i}{n} \right| + \left| Y - \frac{\sum_{i=0}^{n-1} Y_i}{n} \right| \quad (8.6)$$

Kde X, Y – souřadnice kolizního vektoru

X_i, Y_i – souřadnice okolních vektorů (různých od kolizních)

n – počet okolních vektorů.



Snímek 58 Vymazání nesprávného vektoru dle okolí

Situaci schematicky zobrazuje snímek 58. Reprezentanti R_0 a R_3 zde mají kolizní vektory, tedy jeden z nich je chybně. Bod S_{03} je středem úsečky R_0R_3 . Nejbližší 3 reprezentanti k bodu S_{03} jsou R_1 , R_2 a R_4 (reprezentanti R_0 a R_3 nejsou uvažováni), jejichž vektory vstupují do výpočtu hodnoty dS pro oba kolizní vektory. Hodnota dS pro reprezentanta R_3 vychází lépe (nižší číslo), a proto bude jeho vektor ponechán. Reprezentant R_0 a jeho příslušný vektor budou odstraněny z množiny výsledků.

Poznámka: Pokud je kolizních vektorů více, je třeba opět ponechat jen jediný, a to ten s nejmenší hodnotou dS . Všechny ostatní jsou odstraněny z množiny výsledků.

8.3 VEKTOROVÁ MAPA

Výsledkem metody sledování částic je tedy soubor vektorů, přičemž každý je určen konkrétní bublině a popisuje, jak se změnila její poloha za časovou diferenci dt mezi dotyčným a následujícím snímkem. Výsledkem použití vzájemné korelace je však vektorová mapa, tedy přehled rychlostí proudění ve snímku. Následujícím úkolem tedy je docílit stejné vektorové mapy i pro metodu sledování pixelů. Ideálně je žádoucí určit rychlost proudění v libovolném bodě snímku.

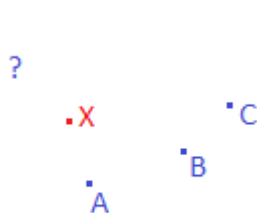
Pokud člověk hodnotí výsledek sledování částic (například snímek 57), pak dokáže svojí inteligencí předpokládat přibližnou rychlost proudění v konkrétním bodě v závislosti na blízkých známých vektorech. Stejnou schopnost lze požadovat i po umělé neuronové síti *MLP*.

8.3.1 Výběr vektorů k učení

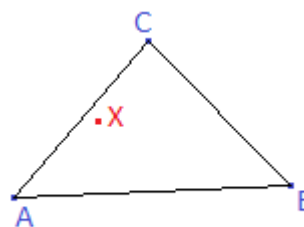
Vektor rychlosti náleží vždy konkrétní bublině, potažmo místu, v němž se tato bublina nachází. Odvození rychlosti bubliny lze dělat na základě osvojení si všech vektorů a jejich závislosti na poloze jim příslušících bublin. Proud však nebývá ideálně plynulý a bubliny se nepohybují po přímkách od vyústky. Tento pohyb je chaotický, obtížně popsatelný a různorodý. Aby si neuronová

síť osvojila veškeré dostupné vektory, musela by mít velký počet neuronů a její učení by pak trvalo velmi dlouho. Inspirací je i v tomto případě člověk. Ani ten se při stanovení rychlosti v konkrétním bodě neřídí všemi dostupnými vektory, ale odhaduje ji na základě pouze několika nejbližších.

První úkolem metody je tedy výběr bublin, na základě kterých se bude neuronová síť rozhodovat. Bublin nemusí být mnoho, ale musí být dostatečně vypovídající. Příklad nevhodně vybraných bublin je na snímku 59a. Body A , B i C leží vzhledem k bodu X relativně stejným směrem, avšak ve výběru není žádná bublina, která by ležela směrem opačným – znázorněno otazníkem. Není jisté, jak se mění rychlost od vybraných bublin směrem k bodu X , neboť není zřejmé, jak má rychlost vypadat za bodem X . První a nejrychlejší možností výběru vektoru k učení je vybrat pro jistotu větší počet blízkých bublin (například 10), avšak ani tehdy není zaručeno, že nebudou nastávat situace jako je ta na snímku 59a. Snímek 59b naproti tomu zachycuje vhodně vybranou hranici. Trojúhelník, jehož vrcholy ABC tvoří reprezentanti vybraných bublin, obsahuje vyšetřovaný bod X . Lze říci, že rychlost je možné správně odvodit ve všech bodech uvnitř dotyčného trojúhelníka.



a) nevhodně vybrané bubliny



b) vhodně vybrané bubliny

Snímek 59 Výběr bublin pro učení dotyčných vektorů

V procesu hledání takového trojúhelníka je nejprve třeba seřadit všechny uvažované reprezentanty podle jejich vzdálenosti od vyšetřovaného bodu X . Lze pro to uvažovat všechny detekované reprezentanty nebo jen k nejbližších. Je však vhodné vždy omezit jejich maximální vzdálenost od vyšetřovaného bodu X . Pokud nelze sestavit žádný takový trojúhelník, pak je rychlost v bodě X stanovena jako nulová.

Pro jednoduchost budou seřazené body označeny pouze indexy, tedy 0, 1, 2 atd. Trojúhelníky jsou řazeny následovně: 210, 310, 320, 321, 410, 420, 421, 430, 431, 432, 510 atd. Základním pravidlem při hledání trojúhelníka je zápis indexů jeho bodů ve tvaru ABC , kdy $A > B > C$. Správně zapsaný trojúhelník je například 831, chybně 230. Prvním uvažovaným trojúhelníkem je 210. Pokud v něm bod X neleží, ověřuje se trojúhelník následující. Jeho výběr je realizován dle následujícího kódu (syntaxe Delphi), v němž proměnné A , B a C představují indexy reprezentantů seřazené vzestupně podle jejich vzdálenosti od bodu X :

```
IF C+1 < B THEN
  C := C + 1
ELSE
  BEGIN
    C := 0;
    IF B+1 < a THEN
```

```

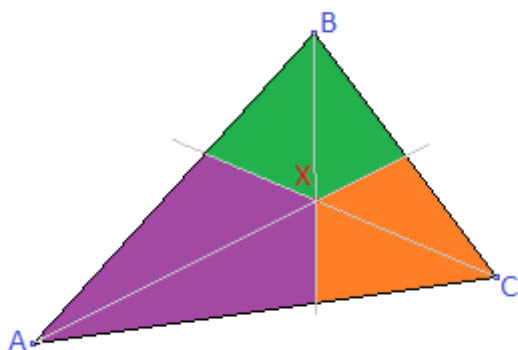
        B := B + 1
ELSE
    BEGIN
        B := 1;
        C := C + 1;
    END;
END;

```

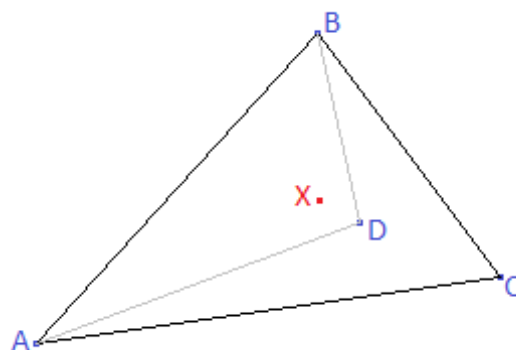
Ověřit, že bod X leží uvnitř konkrétního trojúhelníka, lze například určením vektorů \mathbf{XA} , \mathbf{XB} a \mathbf{XC} a jejich směrových úhlů. Bod X je uvnitř trojúhelníka ABC , pokud platí:

$$|\varphi_{XA} - \varphi_{XB}| + |\varphi_{XB} - \varphi_{XC}| + |\varphi_{XC} - \varphi_{XA}| = 2\pi \quad (8.7)$$

Postup garantuje, že uvnitř trojúhelníku nebude ležet žádný další reprezentant. Takový bod by totiž figuroval v trojúhelníku, který by se uvažoval již dříve a který by bod X rovněž obsahoval. Jinými slovy, pokud trojúhelník ABC obsahuje body X a D , pak jeden z trojúhelníků DAB , DAC , DBC rovněž obsahuje bod X , jako to dokládá snímek 60. Snímek 60a naznačuje rozdělení trojúhelníku přímkami XA , XB a XC . Každému vrcholu bude náležet sektor, sestávající ze dvou trojúhelníků, kterým je dotýčný bod vrcholem. Konkrétně ve snímku 60a náleží bodu A fialově, bodu B zeleně a bodu C oranžově vyplněný sektor. Pokud existuje další bod D uvnitř trojúhelníku ABC , pak nahradí ten vrchol, v jehož sektoru se nachází. Příklad lze vidět na snímku 60b. Bod D se nachází v sektoru bodu C , který proto nahradí. Bod X je tedy rovněž uvnitř trojúhelníka ABD .



a) rozdělení trojúhelníka na sektory



b) nahrazení vrcholu C vrcholem D , který byl právě v sektoru bodu C

Snímek 60 Bublina uvnitř trojúhelníku

Při hledání vhodného trojúhelníka by však trojúhelník ABD byl uvažován dříve, protože bod D je blíže bodu X než bod C . Obecně jakýkoliv bod ($\angle C$) v sektoru bodu C je bodu X blíže než bod C .

Z principu však nemůže existovat trojúhelník s vrcholy v reprezentantech bublin, který by obsahoval bod X , ležící mimo oblast detekovaných bublin. Je však zřejmé, že body mimo proud budou mít nulovou rychlost. Že bod X leží mimo proudnici, lze zjistit právě tak, že neexistuje trojúhelník s vrcholy v reprezentantech platných bublin, který by jej obsahoval. Další možností je například sestavení obálky reprezentantů (z časových důvodů postačuje konvexní, čemuž se věnuje kapitola 6.2). Výhodou je, že pokud je rychlost určována pro více různých bodů (například při tvorbě

vektorové mapy), je třeba obálku bublin tvořit pouze jednou. Další z možností je ověření, že nejmenší kruhová výseč s vrcholem v bodě X obsahující všechny reprezentanty bublin má vnitřní úhel menší než π .

8.3.2 Nastavení neuronové sítě k určení rychlosti v konkrétním bodě

Tomuto problému se věnuje Šťastný a Richter (2016). V předchozí kapitole byla pro bod X nalezena vhodná množina bodů (reprezentantů bublin), z jejichž rychlostí má neuronová síť odvodit rychlost v bodě X . Jako vhodná síť v tomto případě poslouží opět vícevrstvá perceptronová neuronová síť *MLP*, přičemž výstupní hodnotou jejích neuronů bude reálné číslo v intervalu $\langle 0,1 \rangle$.

Od sítě se očekává, že bude rozhodovat o vektoru rychlosti v konkrétním bodě. Bude se učit dle m bodů množiny M , vybraných v kapitole 8.3.1, a jím příslušících vektorů rychlosti. Je však výhodné vstupů i výstupů transformovat na relativní souřadnice:

Transformace souřadnic bodu

Protože je množina M vybrána pro určení rychlosti v konkrétním bodě, neuronová síť nikdy nebude operovat s jinými souřadnicemi, než jaké jsou dány právě body množiny M . Je předpoklad, že vyšetřovaný bod je těmito body obklopen, tedy bude ležet uvnitř jimi vytyčené oblasti. Je tedy zbytečné, aby reagovala na absolutní souřadnice v celém snímku. Pro lepší citlivost vstupů je tedy vhodné je transformovat. Existují dvě možné jednoduché transformace:

- a) **Nejmenší obdélník** – Procházením množiny M je nutno zjistit extrémní hodnoty souřadnic, tedy X_{min} , Y_{min} , X_{max} , Y_{max} . Pokud nebyly tyto body vybrány trojúhelníkovou konstrukcí, je nutné mezi ně zahrnout i vyšetřovaný bod. Extrémní souřadnice přímo definují rozměry obdélníka, jehož hrany budou rovnoběžné se souřadnicovými osami X a Y . Tento obdélník bude zahrnovat všechny body množiny M . Souřadnice $[X_i, Y_i]$ i -tého reprezentanta jsou pak transformovány na souřadnice $[X'_i, Y'_i]$ podle vzorce:

$$X'_i = \frac{X_i - X_{min}}{X_{max} - X_{min}} \quad (8.8)$$

$$Y'_i = \frac{Y_i - Y_{min}}{Y_{max} - Y_{min}} \quad (8.9)$$

Totožnou transformací pak musí projít i vyšetřovaný bod, protože bude předáván síti k vyhodnocení. Všechny transformované body budou mít obě souřadnice v rozmezí $\langle 0, 1 \rangle$. Výhodnou této transformace je velmi malá oblast, na níž bude neuronová síť provádět výpočty.

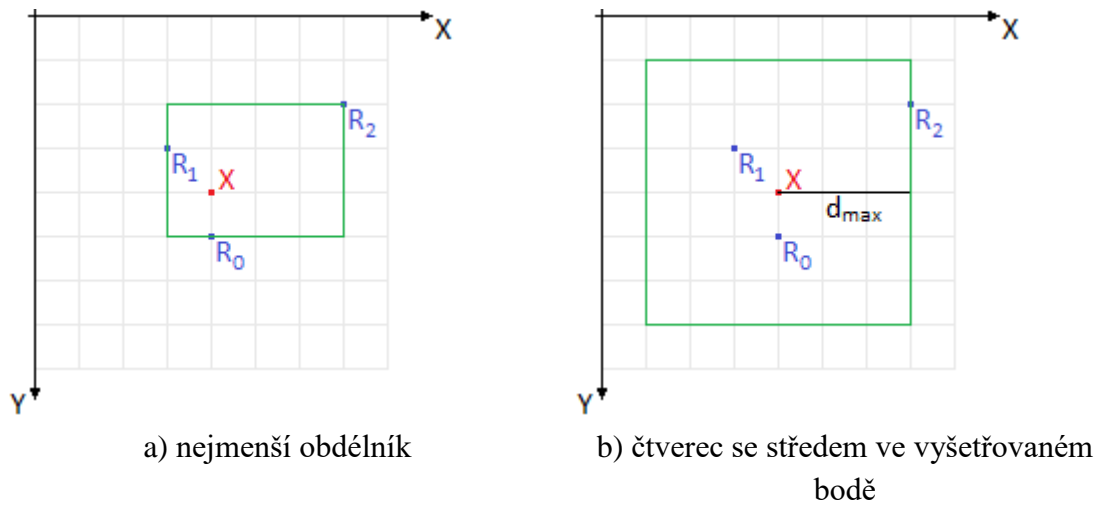
- b) **Čtverec se středem ve vyšetřovaném bodě** – Transformací bude dán nejmenší možný čtverec, jehož hrany jsou rovnoběžné se souřadnicovými osami, který obsahuje všechny body množiny M a v jehož středu se nachází vyšetřovaný bod X o souřadnicích $[X_X, Y_X]$. Je proto třeba určit největší rozdíl souřadnice vyšetřovaného bodu a bodů množiny M , který bude představovat proměnná d_{max} . Souřadnice i -tého reprezentanta $[X_i, Y_i]$ budou transformovány na $[X'_i, Y'_i]$ následovně:

$$X'_i = \frac{X_i - (X_X - d_{max})}{2 \cdot d_{max}} \quad (8.10)$$

$$Y'_i = \frac{Y_i - (Y_X - d_{max})}{2 \cdot d_{max}} \quad (8.11)$$

Souřadnice vyšetřovaného bodu X se transformují na $X_X = 0.5$ a $Y_X = 0.5$. Obecně budou všechny souřadnice dotčených bodů opět v rozmezí $<0, 1>$. Výhodou tohoto přístupu je kromě ideálně umístěného bodu X také čtvercový tvar oblasti, takže obě souřadnice budou mít pro neuronovou síť stejnou váhu.

Princip transformace stejné množiny bodů oběma metodami znázorňuje snímek 61. Nebylo však prokázáno, že by jeden typ transformace byl efektivnější než druhý.



Snímek 61 Transformace vstupu

Transformace očekávaného výsledku

Výstupem neuronové sítě budou dvě reálná čísla v intervalu $<0, 1>$. Souřadnice vzorových vektorů však obecně v tomto intervalu neleží. Procházením množiny vzorových vektorů je třeba vyhledat nejdelší vektor. Teoreticky se jeho směr může ve výsledku změnit, a proto je nutné jeho délku považovat za největší hodnotu, jakou může nabývat souřadnice X nebo Y . Je vhodné tuto velikost ještě vynásobit koeficientem q (například 1.5 nebo 2), čímž je získána maximální hodnota souřadnice C_{max} . Tím je umožněno, aby neuronová síť mohla určit dokonce 1.5krát či 2krát větší souřadnici, než je délka nejdelšího vzorového vektoru. Každá ze souřadnic výsledného vektoru tedy bude moci nabývat hodnot $<-C_{max}, C_{max}>$. Transformace souřadnic vektoru $[dX_i, dY_i]$ na transformované souřadnice $[dX'_i, dY'_i]$ probíhá podle následujících vzorců:

$$dX'_i = \frac{0.5}{C_{max}} \cdot dX_i + 0.5 \quad (8.12)$$

$$dY'_i = \frac{0.5}{C_{max}} \cdot dY_i + 0.5 \quad (8.13)$$

Zpětná transformace obdrženého výsledku

Neuronová síť bude určovat dvojici čísel $\langle 0, 1 \rangle$. Význam těchto hodnot je stejný, jako je význam transformovaných souřadnic vzorových vektorů, které se síť učila. Pro získání reálných souřadnic výsledného vektoru je třeba výsledek neuronové sítě (relativní souřadnice) transformovat zpět na souřadnice absolutní:

$$dX_X = \frac{(dX'_X - 0.5) \cdot C_{max}}{0.5} \quad (8.14)$$

$$dY_X = \frac{(dY'_X - 0.5) \cdot C_{max}}{0.5} \quad (8.15)$$

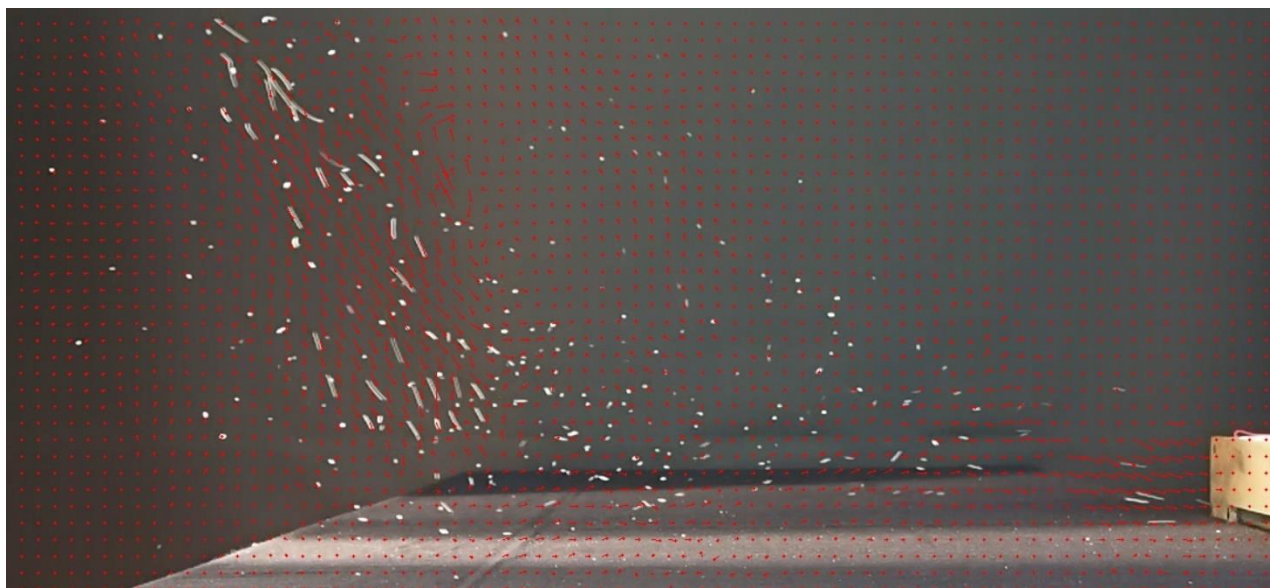
Neuronová síť tedy bude mít dva vstupy a dva výstupy. Pro účely odvození vektorů postačí jediná skrytá vrstva o určitém počtu neuronů. Čím více vzorů má síť pojmout, tím více neuronů by měla obsahovat. Testováním bylo ověřeno, že v případě tří vzorů je dostatečným 10 neuronů skryté vrstvy. Pokud je třeba síť naučit více vzorů, pak pro každý další vzor je vhodné přidat například 2 neurony skryté vrstvy. Síť se vzory učí tak dlouho, dokud je dostatečně přesně nedokáže určit sama. Dle požadované přesnosti to může být cca 50–100 učením na každý vzor.

8.3.3 Tvorba vektorové mapy

Cílem metody je určení vektorové mapy, jako je tomu při použití vzájemné korelace. I v tomto případě je nutné snímek rozdělit na čtvercové podoblasti. V prostředním bodě každé podoblasti je pak nutné určit rychlost proudění, ale nikoliv vzájemnou korelaci, ale postupem zmíněným v kapitolách 8.3.1 a 8.3.2, tedy:

1. Vyhledání bublin, dle jejichž rychlosti se neuronová síť bude učit
2. Transformace souřadnic reprezentantů a vektorů
3. Předání transformovaných vzorů síti + proces učení
4. Předání požadovaného transformovaného bodu síti
5. Převod obdrženého výsledku zpětnou transformací na vektor

Příklad vektorové mapy, získané dotyčným postupem, prezentuje snímek 62.



Snímek 62 Vektorová mapa dle postupu sledování částic vytvořená neuronovou sítí

9 URČENÍ VLASTNOSTÍ VENTILAČNÍHO SYSTÉMU

K nejdůležitějším vlastnostem vyústky patří úhel rozptylu proudu. Úhlem rozptylu je myšlen úhel proudu výstupního vzduchu. Tvarem proudnice je v ideálním případě kuželová plocha. Protože se pracuje s 2D snímky, pak je předpokládáným tvarem výseč, tedy dvojice přímek, která svírá jistý úhel. Protože však má proud omezený dosah, za ideální tvar proudu je považován rovnoramenný trojúhelník. Úhel rozptylu vyústky pak představuje jeho hlavní vrcholový úhel u vyústky. Reálný tvar mlhy v daném okamžiku se však trojúhelníku pouze blíží, jak je patrné například z finálních snímků v kapitolách 5 a 6.

V této kapitole bude popsán postup, jakým jsou významné veličiny proudění stanoveny pro reálný snímek či záznam (sekvenci snímků) proudění vzduchu. Pokud je předpokládáným tvarem proudnice výseč určená dvojicí přímek, je zřejmé, že k určení úhlu rozptylu vyústky je třeba znát právě tuto dvojici přímek. Ve 2D prostoru je lze nazývat „horní“ a „dolní“ přímkou.

9.1 VYŠETŘOVÁNÍ TVARU PROUDNICE V JEDINÉM SNÍMKU

Tato kapitola nastíní výpočty a odvození vybraných veličin v jediném snímku proudění.

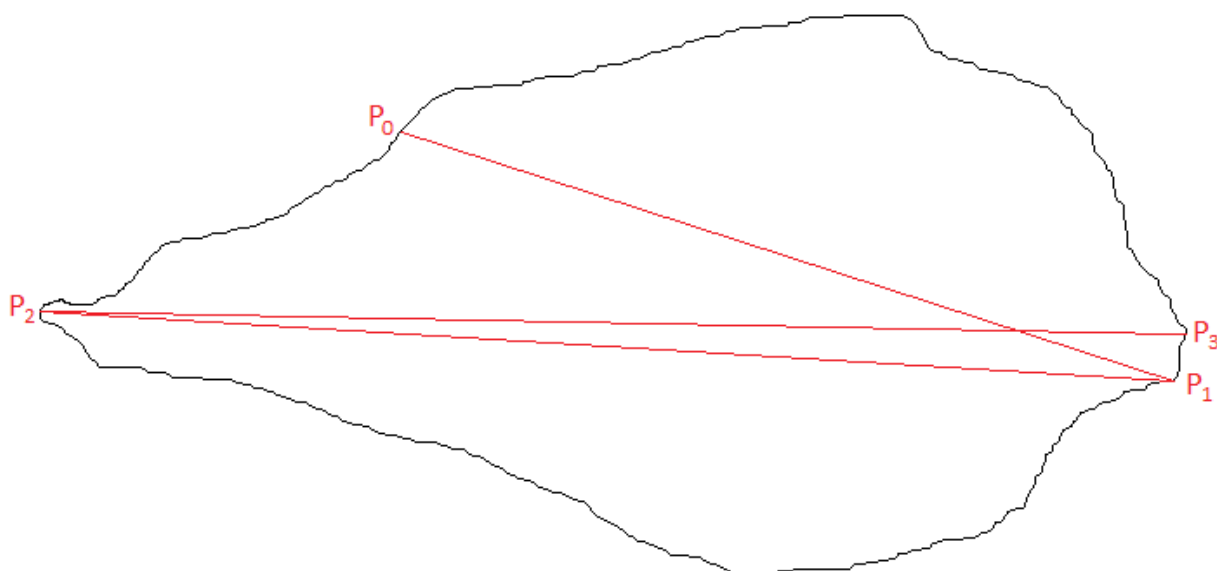
9.1.1 Určení dosahu proudu a hraničních křivek

Jedná se o krok důležitý pro určení dalších atributů proudu. Principem výpočtu je hledání nejdelší možné úsečky uvnitř proudnice mezi dvěma hraničními body. Tuto úsečku lze stanovit úplně přesně časově náročným algoritmem poměrování všech možných úseček mezi všemi hraničními body. Při počtu hraničních pixelů n je to $(n^2 - n) / 2$ úseček.

Efektivnější možností je stanovit ji opakovaným hledáním nejvzdálenějšího bodu pro bod aktuální. Algoritmus spočívá v tom, že pro konkrétní pixel hranice P_0 (na počátku zvolen libovolně) jsou procházeny úsečky se všemi hraničními pixely. Z těch je vybrán pixel P_1 , který je od pixelu P_0 nejdále. Pro dosažený pixel P_1 je pak hledán stejným způsobem pixel P_2 . Algoritmus končí v případě, že nejvzdálenějším pixelem pixelu P_x je předchozí uvažovaný pixel P_{x-1} . Stejný efekt bude mít podmínka, že každá nová úsečka (P_i, P_{i+1}) musí být delší než předchozí (P_{i-1}, P_i) . Tato podmínka je sama o sobě dostatečná, a navíc chrání algoritmus proti zacyklení, které by mohlo nastat například u oblasti ve tvaru rovnostranného trojúhelníku.

Algoritmus je patrný ze snímku 63, kde bod P_0 je náhodně zvolený bod hranice. V tomto případě je nejdelší nalezená úsečka mezi (P_2, P_3) . Pixely jsou označeny P_A a P_B a samotná úsečka je prohlášena za dosah proudu d .

Dosah proudu může být přibližně určen také ze známé polohy vyústky (stává se bodem P_A) a z jí nejvzdálenějšího bodu vektorové mapy (stává se bodem P_B), který má zvolenou dostatečně velkou rychlost.



Snímek 63 Hledání nejdelší úsečky v ohraničené oblasti – nalezena (P_2, P_3)

Úsečka P_AP_B pŕl proud na dvě podoblasti a lze ji připodobnit k výšce náležící základně trojúhelníku ideální proudnice. Úsečka d tak rozděluje hranici na dvě části. V následujícím textu bude uvažován horizontální směr proudění, které bylo snímáno z bočního pohledu. Proto budou získané části hranice označovány jako horní a dolní hraniční křivka.

Poznámka: Některé metody určení tvaru proudnice přímo generují dvojici hraničních křivek, jako například detekce proudu v řezech, o níž pojednává kapitola 5.2. Dosah proudu je stanoven stejně, jako bylo popsáno výše.

Poznámka: Přímkou danou body P_A, P_B nelze považovat za přesnou osu proudu.

9.1.2 Určení polohy vyústky

Poloha vyústky P_S je většinou uživateli známá. Ten ji může předat pro další zpracování. Pro proudění bez turbulencí či dalších okolních vlivů ji však lze dopočítat z hranice a dosahu proudu.

Počátečním pixelem proudění P_S (pozice vyústky) je jeden z bodů P_A, P_B náležících úsečce dosahu d . K rozhodnutí, který z obou bodů budeme pokládat za počátek proudu, může posloužit několik postupů. Všechny vychází z předpokládaného ideálního tvaru proudu, tedy rovnoramenného trojúhelníku. Pro stanovení počátku proudění lze aplikovat více různých přístupů zároveň a poté se přiklonit k nejčastějšímu výsledku.

- a) **Určení počátku dle kolmice** – Tento přístup spočívá v hledání úsečky, představující fiktivní základnu rovnoramenného trojúhelníka ideální proudnice. Tato základna musí být kolmá na úsečku (P_A, P_B) . Úsečka (P_A, P_B) bude představovat výšku hledané základny. Posupuje se od bodu P_A k bodu P_B , přičemž pro každý pixel P_i je sestrojena přímka kolmá na (P_A, P_B) . Na této přímce jsou nalezeny všechny hraniční pixely (obvykle 2) a vybrána nejkratší úsečka U_i , která všechny tyto body obsáhne. Procedura vede k určení bodu P_k , jehož úsečka U_k je ze všech úseček U_i nejdelší. Protože v trojúhelníku platí, že se hlavní vrchol nachází na opačné straně jako základna, je za vrchol (a počátek proudu) prohlášen ten z pixelů P_A, P_B , který je od pixelu P_k vzdálenější.

- b) **Určení počátku dle vzdálenosti hraničních bodů** – Z tvaru trojúhelníka vyplývá, že vzdálenost pixelů od hlavního vrcholu je v součtu vyšší než jejich vzdálenost od paty výšky. Body trojúhelníka jsou blíže patě výšky než hlavnímu vrcholu. Tento přístup tedy počítá vzdálenosti všech hraničních pixelů od P_A resp. P_B a obě hodnoty porovná. Vyšší z obou součtů náleží počátku proudění.
- c) **Určení počátku proložením hraničních křivek** – Úsečka (P_A, P_B) rozděluje hranici na horní a dolní hraniční křivku. Obě křivky jsou proloženy přímkami (například lineární regresí pro metodu nejmenších čtverců – kterou využívá například Freedman (2005)) a je určen průnik obou přímek. Počátkem proudění je ten z pixelů P_A, P_B , který je tomuto průniku blíže.

9.1.3 Výběr reprezentantů hranice

Počátek proudění P_S byl detekovaný jako jediný pixel. V reálu není příliš pravděpodobné, aby průměr proudu vzduchu u vyústky byl roven jedinému pixelu. Naopak je odvislý právě od plochy vyústky. Proto je třeba pro vyhodnocování hranice blízké okolí pixelu P_S vyloučit. Stejně tak příliš vzdálené pixely od P_S již nemusí mít vypovídající hodnotu o vlastnostech vyústky. Při určování rozptylu se tedy budeme soustředit pouze na pixely v jistém rozmezí vzdáleností od bodu P_S .

Množinu pixelů, dle kterých budeme určovat vlastnosti vyústky, budeme nazývat množinou reprezentantů horní resp. dolní hranice. Tou budou buď všechny pixely na horní resp. dolní hranici v konkrétním rozsahu vzdálenosti od P_S , anebo je jejich počet dále zredukován. Kompletní sestavení zredukované množiny reprezentantů bude demonstrováno na uspořádané množině n pixelů horní hranice, kdy P_0 odpovídá pixelu P_S a P_n je od něj nejvzdálenějším bodem hranice (druhým pixelem z dvojice P_A, P_B).

Prvním reprezentantem horní hranice je první pixel v pořadí, splňující podmínku minimální vzdálenosti od P_S . Každý následující reprezentant je vždy vybrán jako první v pořadí, jehož vzdálenost od předchozího překoná předem zvolenou konstantní hodnotu vzdálenosti mezi reprezentanty. Zároveň však nesmí překonat hodnotu maximální vzdálenosti od P_S .

Redukce množiny reprezentantů není povinná. V opačném případě jsou reprezentanty všechny pixely mezi prvním, který splňuje podmínku minimální vzdálenosti od P_S (včetně) a prvním, který nesplňuje podmínku maximální vzdálenosti (vyjma). Analogický postup je aplikován na dolní hranici. Výsledkem tohoto kroku jsou tedy dvě množiny reprezentantů – pro horní a dolní hranici.

9.1.4 Hraniční přímky, osa proudu a rozptyl

Horní hraniční přímka p_H je získána lineární regresí množiny reprezentantů horní hranice (o čemž pojednává například Freedman (2005)). Stejně je postupováno i v případě dolní hraniční přímky p_D .

Pro určení dalších parametrů je důležité sjednotit vyjádření sklonu obou hraničních přímek. Například přímka se sklonem $\pi/6$ má zároveň sklon i $7\pi/6$. Úhly obou hraničních přímek by měly být v totéž rozmezí, tedy například $(-\pi/2, \pi/2)$ nebo $(\pi/2, 3\pi/2)$. Správně stanovené úhly jsou například $\varphi_H = -\pi/6$, $\varphi_D = \pi/6$ nebo $\varphi_H = 5\pi/6$, $\varphi_D = 7\pi/6$. Nevhodně by úhly byly stanoveny například $\varphi_H = -\pi/6$, $\varphi_D = 7\pi/6$ nebo $\varphi_H = 5\pi/6$, $\varphi_D = \pi/6$.

Z matematického vyjádření obou hraničních přímek lze dále určit osu proudění. Ta bude procházet průsečíkem obou přímek. Její úhel sklonu je roven průměru úhlu sklonů obou hraničních přímek. Ze známých sklonů hraničních přímek je možné určit další neznámou veličinu proudění,

a to rozptyl vyústky. Úhel rozptylu je dán jako $|\varphi_H - \varphi_D|$. Absolutní hodnota je použita pro případ, že by hodnota úhlu φ_H byla menší než hodnota úhlu φ_D .

9.2 VYŠETŘOVÁNÍ TVARU PROUDNICE V SEKVENCI

V kapitole 3.4 bylo zmíněno možné použití množiny snímků téhož proudění ke zpřesnění hodnot detekovaných vlastností o tomto proudění. Pokud bude proudění hodnoceno podle jediného jeho snímku, není zaručeno, že dotyčný snímek má dostatečnou vypovídací hodnotu. Naopak, čím větší je zpracované množství snímků stejného proudění, tím vyšší vypovídací hodnotu bude souhrnný výsledek mít.

9.2.1 Statistické zpracování výsledků jednotlivých detekcí

Pro tento postup není důležitá časová známka snímku. Proto bývá sekvence pořízena opakovaným focením. Lze však vyhodnocovat i videozáznam, z něhož se jednotlivé snímky vyseparují.

Metoda se používá především ke stanovení rozptylu vyústky. Ten je nejprve určen pro každý snímek sekvence. Určení rozptylu vyústky v konkrétním snímku je popsáno v kapitole 9.1.4. Celý proces lze souhrnně zapsat následujícími kroky:

1. Detekce oblasti proudnice.
2. Určení horní a dolní hranice proudu (pokud není určena přímo detekcí proudnice v řezech).
3. Výběr množiny reprezentantů horní a dolní hranice.
4. Aproximace reprezentantů přímkami pro horní a dolní část hranice.
5. Stanovení osy proudu a rozptylu vyústky.

Výše popsané vyhodnocení je provedeno pro každý snímek sekvence. Jednotlivá data jsou pak zpracovávána statisticky.

Statistické vyhodnocení úhlů.

Úhly představují sklony hraničních přímek a osy proudu. Při výpočtu průměrného úhlu osy v sekenci do statistického vyhodnocení vstupují úhly sklonu osy detekované v každém snímku sekvence.

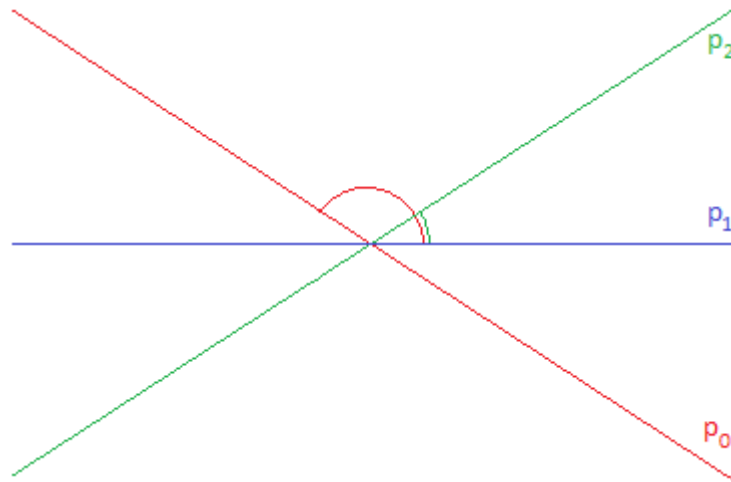
Pro sklon přímky α obecně platí, že je ekvivalentní všem hodnotám $\alpha + k \cdot \pi/2$, kde k je libovolné celé číslo. Například průměr úhlů $-\pi/6, 0, \pi/6$ je 0. Ale pokud je úhel $-\pi/6$ nahrazen úhlem $5\pi/6$ (z hlediska sklonu osy ekvivalentní), výsledný průměr bude $\pi/3$. Pokud je tentýž úhel zaměněn za ekvivalentní úhel $11\pi/6$, výsledný průměr bude $2\pi/3$.

Uvedený příklad vede k nutnosti převodu směrových úhlů na úhly relativní. Pro každou přímku je nutné vyhodnotit, jaký úhel α_i by měla kruhová výseč, obsahující všechny přímky v případě, že by byla z jedné strany ohraničená právě touto úsečkou. Pro každou přímku je hledán nejmenší kladný úhel, jenž by obsáhl všechny úhly, které svírá dotyčná přímka se všemi ostatními přímkami.

Příklad:

Sklony přímek $\varphi_0 = -\pi/6$; $\varphi_1 = 0$, $\varphi_2 = \pi/6$. Například výseči pro přímku p_1 , která pojme všechny ostatní sklony přímek, bude úhel určen jako větší z kladných úhlů $\{\varphi_0 - \varphi_1; \varphi_2 - \varphi_1\}$, tedy větší

z hodnot $\{5\pi/6 - 0; \pi/6 - 0\}$. Nejmenší kruhová výseč pro přímkou p_1 je $5\pi/6$, ve snímku 64 červeně – v červené výseči $0-5\pi/6$ jsou obsaženy sklony všech tří přímek.



Snímek 64 Hledání nejmenší kruhové výseče s hranicí p_1 danou sklonem 0, který obsahuje všechny detekované přímky.

Pro přímkou p_0 je hledána vyšší z hodnot $\{\varphi_1 - \varphi_0; \varphi_2 - \varphi_0\} = \{0 + \pi/6; \pi/6 + \pi/6\}$, což je $\pi/3$. Pro přímkou p_2 je hledána vyšší z hodnot $\{\varphi_0 - \varphi_2; \varphi_1 - \varphi_2\} = \{5\pi/6 - \pi/6; \pi - \pi/6\}$, což je $5\pi/6$. Nejvyšší z výsledných úhlů je úhel $\pi/3$ výseče pro přímkou p_0 . Obrácená hodnota jejího sklonu $-\varphi_0 = \pi/6$ definuje transformaci. Absolutní úhly $\varphi_0 = -\pi/6; \varphi_1 = 0, \varphi_2 = \pi/6$ jsou transformovány na $\varphi'_0 = 0; \varphi'_1 = \pi/6, \varphi'_2 = \pi/3$. Tyto úhly lze zprůměrovat ($= \pi/6$) a výsledek převést reverzní transformací ($-\pi/6$) do absolutních souřadnic. Průměrný sklon této množiny přímek je 0.

Statistické vyhodnocení přímek

Cílem tohoto statistického vyhodnocení je získání přímky, která by vhodně reprezentovala množinu přímek stejného charakteru, z nichž každá pochází z jednoho ze snímků sekvence. Nejčastěji je určena reprezentační přímka pro osu snímku, horní hranici a spodní hranici.

Reprezentační přímku lze určit výpočtem jejího sklonu a stanovením bodu, kterým prochází. Výpočtu sklonu se věnuje předchozí kapitola Statistické vyhodnocení úhlů. Bod, kterým výsledná přímka prochází, je dán průměrováním X a Y souřadnic všech možných průniků jednotlivých přímek. Pokud jsou všechny zdrojové přímky navzájem rovnoběžné, bude s nimi rovnoběžná i přímka výsledná. Bod, kterým prochází, bude určen jako průměr průsečíků přímek s jejich kolmicí.

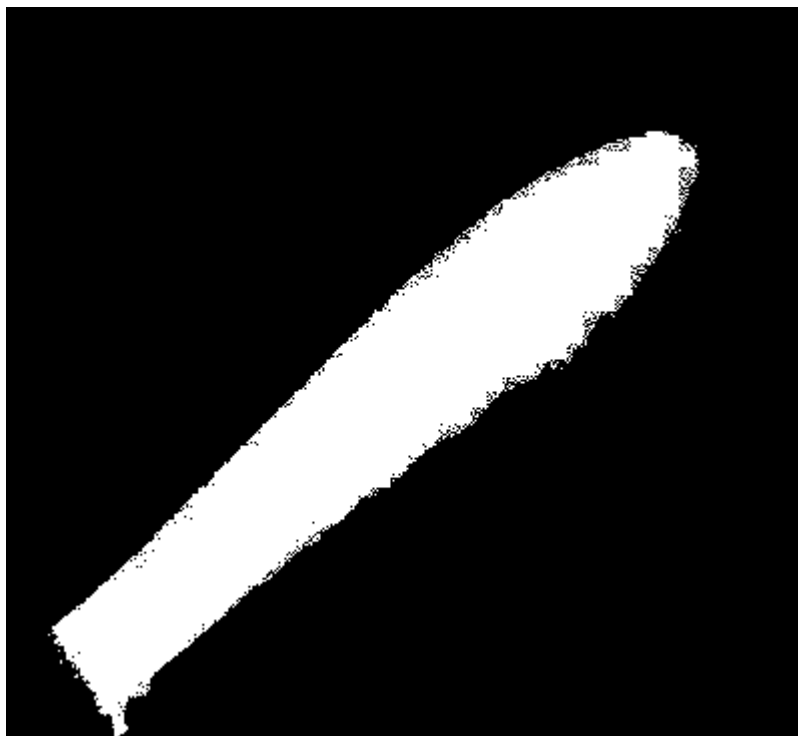
9.2.2 Sčítání detekovaných pixelů

V předchozích kapitolách bylo uvedeno několik způsobů, jakými lze detekovat, zda konkrétní pixel náleží proudnici či jejímu okolí. Metody jako „prahování“ či „detekce mlhy neuronovou sítí“ takovou informaci přímo generují. Metody „Vyšetřování proudnice v řezech“ či „Detekce nekonvexního ohraničení částic“ tvoří hranici jiným principem, avšak pro každý pixel lze pak usoudit, jestli je uvnitř hranice, a tedy je pixelem proudu, nebo je vně hranice a pak je pixelem okolí.

Jestliže lze pro každý snímek sekvence určit, které pixely náleží proudu a které nikoliv, je možné sekvenci podrobit procesu sčítání detekovaných pixelů. Pro tuto metodu zpracování sekvence snímku není nutná časová známka jednotlivých snímků. Jejím principem je tvořit z výsledků detekce

proudu v jednotlivých snímcích jeden souhrnný výsledek. Pro každý pixel je napočítáváno, v kolika snímcích sekvence byl označený jako pixel proudu. Dále je stanovena hraniční hodnota Q procent. Pokud je konkrétní pixel alespoň v Q procentech snímků sekvence vyhodnocen jako pixel proudu, pak je pak považován za skutečný pixel proudu. Výsledkem celého vyhodnocení je tedy jediný souhrnný snímek, jenž může být dále podroben například vyhodnocení, popsanému v kapitole 9.1 či jiným.

Zvolená konstanta $Q > 0$ však může způsobit také nežádoucí jev, a to označení osamocených pixelů jako pixely mlhy. Tento jev je patrný ze snímku 65, kdy některé pixely na hranici podmínku splnily, ale jejich sousední již nikoliv. Takovému jevu lze předejít aplikací vhodného vyhlazovacího mechanismu. Například kombinací filtru typu dolní propust a prahování. Na snímku 66 je výsledek po filtraci maticí 5×5 , kdy každý její prvek má hodnotu $1/25$. Na snímku 67 je pak výsledný produkt po opětovném prahování prahovou intenzitou 40. Po filtraci budou mít opravdu osamocené pixely (mimo proud) hodnotu $255 / 25 = 10.2$, tedy následné prahování je označí jako okolí. Podobně jsou eliminovány pixely poblíž proudnice, které měly původně hodnotu 0, ale filtrací se jim zvýšila.



Snímek 65 Nespojité hranice po sčítání pixelů



Snímek 66 Lineární filtrace snímku 65



Snímek 67 Prahování snímku 66 s prahem 40

9.3 VYHODNOCENÍ RYCHLOSTI

Kapitoly 7 a 8 popisují určení rychlosti v konkrétním bodě, které je dále používáno při konstrukci vektorové mapy. Ta poslouží jako základ pro následující metody.

9.3.1 Validace vektorové mapy

Jedná se o základní úpravu vektorové mapy, pro kterou je využita detekovaná proudnice v tomto snímku. Nenulový vektor rychlosti ve vektorové mapě může být platný pouze tehdy, pokud jeho bod leží uvnitř proudnice. V opačném případě je dotyčnému bodu namísto detekovaného vektoru přiřazen vektor $[0, 0]$. Touto triviální metodou lze zkombinovat výsledky detekce tvaru a detekce rychlosti, což vede k eliminaci chybných vektorů, které leží mimo proudnici.

9.3.2 Filtrace vektorové mapy

Vektorovou mapu lze dále zpracovávat například vektorovými filtry. Ty pracují podobně jako filtry obrazové, pouze s tím rozdílem, že filtrují zvlášť každou souřadnici. Pokud filtrační maska má charakter dolní propusti tedy

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad (9.1)$$

pro prostřední bod oblasti s vektory rychlosti

$$\begin{bmatrix} [3, 1] & [2, 0] & [4, 0] \\ [-2, -4] & [-3, 3] & [6, 1] \\ [3, -2] & [2, 3] & [3, -2] \end{bmatrix} \quad (9.2)$$

budou filtrace určeny souřadnice vektoru rychlosti

$$dX = (3 + 2 + 4 - 2 - 3 + 6 + 3 + 2 + 3) / 9 = 2$$

$$dY = (1 + 0 + 0 - 4 + 3 + 1 - 2 + 3 - 2) / 9 = 0$$

Vektorová filtrace přináší podobný efekt jako filtrace obrazová. Například filtrace typu dolní propust pohlcuje případné chyby a zahlučuje změny mezi vektory rychlostí blízkých bodů. Filtrace typu horní propust naopak zvýrazňuje rozdíly mezi okolními vektory. Výrazný rozdíl mezi rychlostmi blízkých bodů však bývá způsoben spíše chybou ve vyhodnocovacím procesu, a proto je výhodné používat především filtr typu dolní propust, po jehož aplikaci bude vektorová mapa plynulejší a věrohodnější.

9.3.3 Zpracování rychlosti v sekvenci snímků

Podobně jako v případě vyhodnocování tvaru proudnice, může sekvence snímků sloužit i ke statistickému zpracování rychlosti. Pokud je vektorová mapa tvořena pro n snímků sekvence, a to ve stejných bodech snímku, pak každý tento bod může výslednou rychlost vyvodit z jeho rychlostí v jednotlivých snímcích.

Poznámka: V tomto případě není nutné, aby snímky následovaly jeden po druhém. Mohou být ze sekvence vybrány libovolně a v libovolném pořadí.

- **Průměrování rychlosti** – Jestliže má konkrétní bod v i -tém z n snímků sekvence vektor rychlosti $[dX_i, dY_i]$, pak souřadnice výsledného vektoru budou průměry n hodnot konkrétních souřadnic. Tedy

$$dX = \frac{\sum_{i=0}^{n-1} dX_i}{n} \quad (9.3)$$

$$dY = \frac{\sum_{i=0}^{n-1} dY_i}{n} \quad (9.4)$$

- **Modus, medián** – Existují další možnosti, jak statisticky zpracovat množinu souřadnic dX_i a dY_i . Lze tak pro výsledný vektor vybrat nejčtenější hodnoty těchto souřadnic či prostřední hodnoty těchto souřadnic.
- **Vyloučení chybných vektorů** – Chybný vektor lze naléznout například tak, že je alespoň jedna jeho souřadnice příliš rozdílná od průměru. Vektory téhož bodu tedy musí splňovat specifickou podmínku, kterou lze vyjádřit buď absolutně

$$|dX_i - \overline{dX}| < k \quad (9.5)$$

$$|dY_i - \overline{dY}| < k \quad (9.6)$$

či relativně

$$\frac{|dX_i - \overline{dX}|}{\overline{dX}} < q \quad (9.7)$$

$$\frac{|dY_i - \overline{dY}|}{\overline{dY}} < q \quad (9.8)$$

Kde k – přílišná absolutní odchylka od průměru

q – přílišná relativní odchylka od průměru

\overline{dX} – průměr dX_i souřadnic vektorů

\overline{dY} – průměr dY_i souřadnic vektorů

10 ZHODNOCENÍ A VÝSLEDKY PRÁCE

V této práci jsou zpracovávány snímky a záznamy (sekvence snímků) proudění vzduchu, vizualizované buď mlhou nebo heliovými bublinami. Hlavními sledovanými veličinami jsou tvar proudnice a rychlost proudění.

10.1 TVAR PROUDNICE

Tato kapitola bude hodnotit jednotlivé metody detekce tvaru proudnice, popisovat jejich využití, přednosti a negativa.

Pro snímky proudění vizualizovaného pomocí mlhy byly zmíněny tyto metody:

- **Prahování** – Jde o velmi rychlou a elementární metodu. Prahování je však vhodné pouze pro snímky prvního typu. Složitější snímky si většinou žádají upravení prahu, aby detekce nedetekovala body okolí. To však s sebou nese i vyloučení méně husté mlhy z proudu. Pro snímky třetího typu už však prahování není použitelné vůbec.
- **Podmíněné prahování** – Oproti běžnému prahování lze podmíněné prahování použít i na některé snímky druhého typu. Vyhodnocení je sice pomalejší než v předchozím případě, ale stále velmi rychlé.
- **Vyšetřování proudnice v řezech** – Jedná se o metodu použitelnou pouze ve snímcích prvního typu. Její výhodou je možnost volby vzdálenosti jednotlivých řezů, což může znamenat úsporu rychlosti za cenu menšího rozlišení výsledku. Výsledky lze však přímo předat metodám pro určení vlastností ventilačního systému, a to bez nutnosti sestavení celkové obálky proudnice.
- **Základní zpracování snímku s využitím neuronové sítě** – Jedná se o metodu výborně fungující pro snímky prvního a druhého typu. Dobře funguje i pro snímek třetího typu, avšak může se stát, že za mlhu bude označeno místo okolí barevně velmi podobné mlze (například světla a odlesky). Její nevýhodou může být nutnost vytvoření a učení neuronové sítě. Avšak tento proces je nutný provést pro celou sekvenci pouze jednou.
- **Podmíněné prahování spojitě detekce** – Jedná se o vylepšení základního zpracování neuronové sítě, v rámci něhož se oblast mlhy postupně rozšiřuje do méně zřejmých míst. Detekovaná hranice proudnice je ještě o něco kvalitnější. I pro tuto metodu však platí, že některá místa okolí mohou být chybně označena jako mlha. Těchto míst je však o něco méně, neboť horní práh pro body, které jistě náleží mlze, bývá nastaven přísněji. Oproti základnímu zpracování neuronovou sítí dochází také k mírnému navýšení procesního času, které je však v porovnání se zbytkem procesu zanedbatelné.
- **Metoda dvou expertů** – Tato metoda je velmi dobře účinná pro řešení chybně detekovaných oblastí ve snímku třetího typu při detekci jedinou neuronovou sítí. Z principu plyne, že každý pixel snímku je vyhodnocován dvěma neuronovými sítěmi, tedy rychlost procesu je dvojnásobná oproti základnímu zpracování neuronovou sítí. Další nevýhodou je komplikovanější proces učení, zejména v případě specifického experta. I v tomto případě platí, že proces učení je třeba vykonat pouze jednou pro celou sekvenci (avšak pro dvě neuronové sítě).

- **Podmíněná detekce dvojicí expertů – Podmíněné prahování jednotlivých výsledků** – Tato možnost představuje velmi kvalitní zpracování všech typů snímků. Přínos podmíněného prahování však není tak výrazný. Nevýhody metody jsou tytéž jako v případě metody dvou expertů. Nárůst procesního času oproti zmíněné metodě je zanedbatelný.
- **Podmíněná detekce dvojicí expertů – Podmíněné prahování sloučeného výsledku** – Jedná se o nejlepší nalezenou metodu, použitelnou pro všechny typy snímků. Jejimi nedostatky jsou opět složitější proces učení neuronových sítí a delší procesní čas.

Pro snímky prvního typu lze obecně doporučit metodu prahování či podmíněného prahování. Pokud má detekce sloužit ke stanovení úhlu rozptylu vyústky, pak je plně dostačující i vhodná detekce proudnice v řezech. Pro snímky druhého typu je plně dostačující metoda základního zpracování snímku s využitím neuronové sítě, případně její rozšíření podmíněným prahováním. Snímky třetího typu je vhodné vyhodnocovat metodou dvou expertů, případně podmíněnou detekcí dvojicí expertů. Na snímku 68 je proudnice detekovaná nejlepší nalezenou metodou, a to podmíněným prahováním sloučení výsledků detekcí dvojicí expertů (viz podkapitola Podmíněné prahování sloučeného výsledku v kapitole 5.3.6).

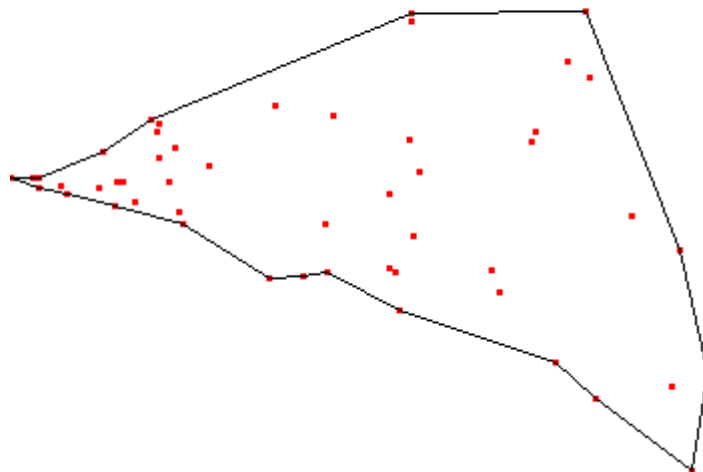


Snímek 68 Určení tvaru proudnice metodou podmíněné detekce dvojicí expertů

Snímky proudění vizualizovaného pomocí heliových bublin jsou obecně méně vhodné pro detekci tvaru proudnice. Jedním z dílčích cílů práce bylo vyvinout metodu, která by byla kvalitativně srovnatelná s výsledky vizualizace pomocí látek, tvořících souvislá vlákna. V práci byly zmíněny tyto metody:

- **Konvexní obálka** – Pro účely detekce tvaru proudnice je tato metoda nedostatečná. Lze ji použít pouze omezeně na malé množství bublin blízko vyústky, neboť vzdálenější bubliny by mohly nežádoucím způsobem deformovat proudnici.
- **Nekonvexní obálka s využitím genetických algoritmů** – Tato metoda generuje kvalitativně velmi podobnou obálku, jako metoda vyšetřování proudnice v řezech. Nekonvexní obálka je tedy věrohodná a použitelná pro další výzkum charakteristik systému. Nevýhodou této metody je její velká časová náročnost a větší chybovost pro

množiny bodů větší než cca 80 bodů. Příklad takto detekované nekonvexní obálky je na snímku 69.



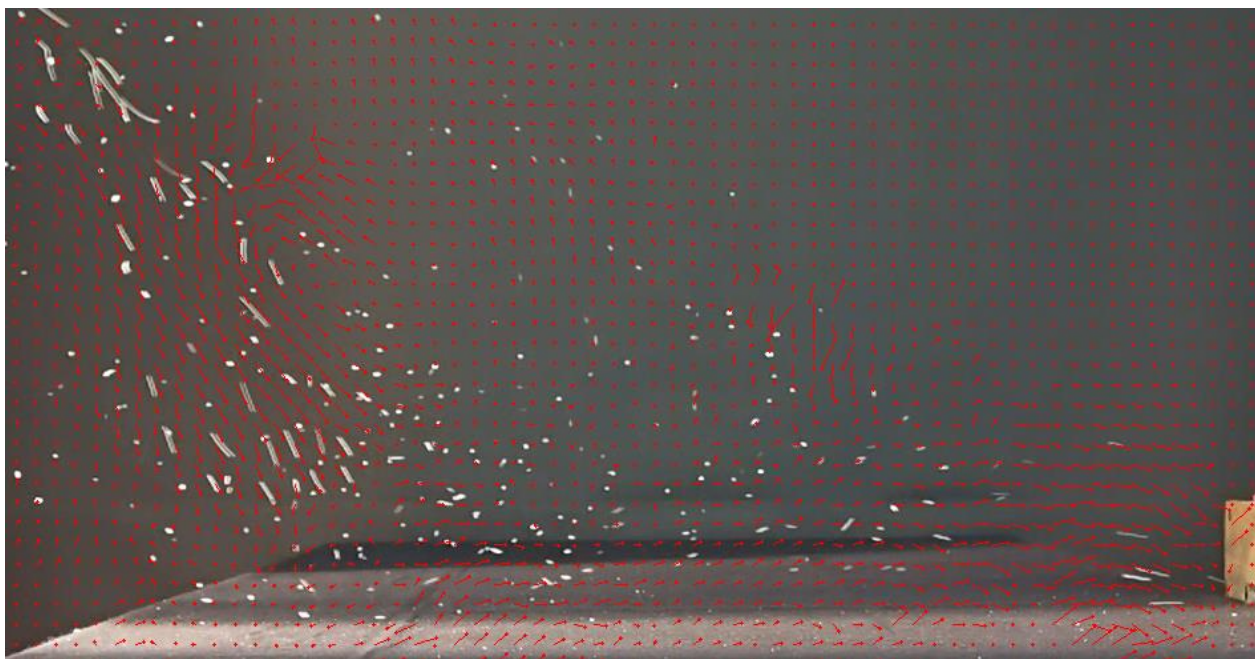
Snímek 69 Nekonvexní ohraničení bodů s využitím genetických algoritmů (50 bodů)

10.2 RYCHLOSTI PROUDĚNÍ

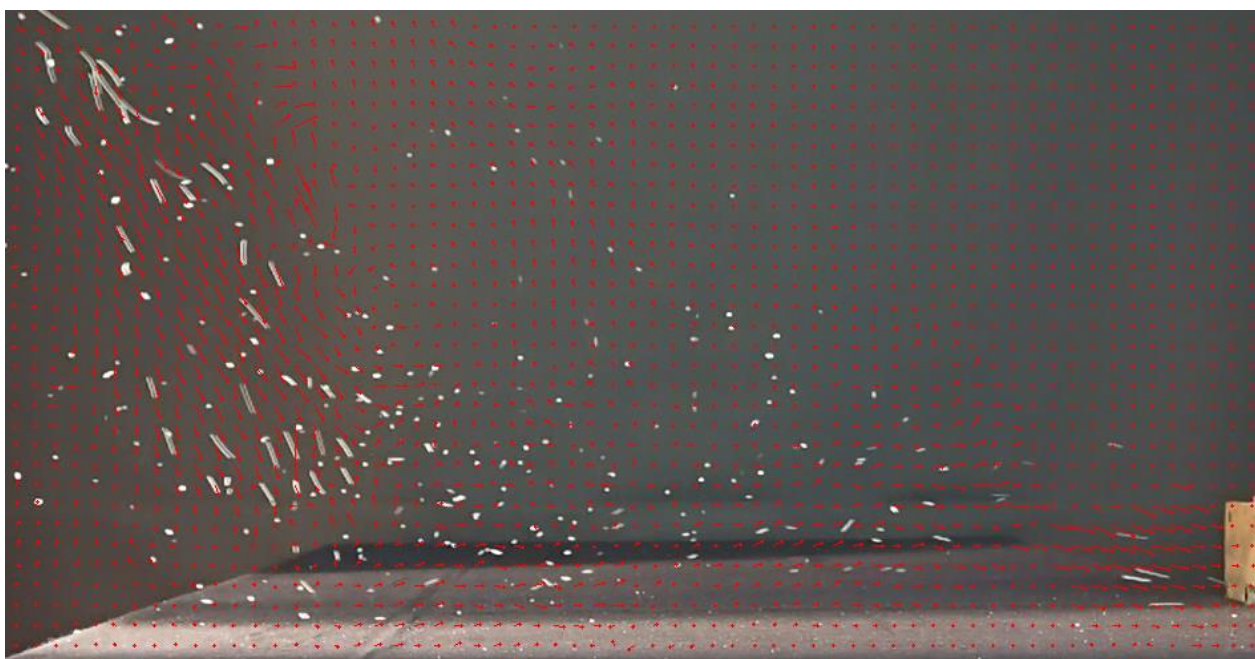
V této práci jsou zmíněny dvě metody pro vyhodnocení rychlosti proudění v libovolném bodě, které se tak hodí i k sestrojení vektorové mapy:

- **Použití vzájemné korelace** – Jde o metodu pro sekvence prvního typu, která je vhodná pro proudy vizualizované látkami tvořícími souvislá vlákna. Použitelná je i pro vizualizaci látkami, které souvislá vlákna netvoří, pokud je jejich hustota v proudu dostatečně velká. Tato metoda nebyla vyvinuta v rámci této práce. Práce ji však rozšiřuje o některá vylepšení.
- **Sledování částic** – Pokud je proudění vizualizováno látkami netvořícími souvislá vlákna a pokud je hustota těchto částic v proudu malá, je vhodné použít metodu sledování částic. V rámci této metody pak o vektoru rychlosti v konkrétním bodě rozhoduje neuronová síť. Metoda operuje s konkrétními bublinami, tedy pokud bude existovat možnost detekce částic ve snímcích druhého či třetího typu, je metoda sledování částic použitelná i pro ně. Výhodou metody je také její použitelnost v případech, kdy se ve 2D snímku v místě o podobných souřadnicích $[X, Y]$ pohybují bubliny různými rychlostmi (způsobeno jejich různými souřadnicemi Z).

Metoda sledování částic doplňuje metodu použití vzájemné korelace a nahrazuje ji při použití na specifický typ proudění. Na snímcích 70 a 71 je porovnání vektorových map vytvořených použitím vzájemné korelace (snímek 70) a metodou sledování částic (snímek 71).



Snímek 70 Vektorová mapa vytvořená použitím vzájemné korelace
Zdroj: Šťastný, Richter, 2014



Snímek 71 Vektorová mapa vytvořená metodou sledování částic
Zdroj: Šťastný, Richter, 2014

11 ZÁVĚR

Při návrhu ventilačního systému hrají významnou roli metody vizualizace proudění. Ty pomáhají vykreslovat proudnice, aby bylo patrné, jak se vzduch vycházející z vyústky systému šíří v místnosti. Tato práce se zabývá možnostmi, jak snímky či záznamy proudění takto vizualizovaného vzduchu počítačově zpracovat.

Práce se soustředí na dva typy vizualizačních metod. Prvním typem je vizualizace látkami tvořícími souvislá vlákna. Takovými látkami mohou být například mlha či kouř. Dalším typem je vizualizace látkami, které netvoří souvislá vlákna, ale částice. Konkrétně jsou v práci vyhodnocovány snímky proudění vzduchu vizualizované heliovými bublinami. Obdobnými aditivy mohou být například saze nebo jiskry.

Hlavními veličinami, které jsou v práci určovány, jsou tvar proudnice a rychlost proudění. Z těchto dat jsou pak definovány další charakteristiky proudění a celého ventilačního systému. Cílem přitom bylo, aby pro obě zmíněné vizualizační metody existovaly postupy, jak z nich požadované veličiny určit.

Snímky a záznamy proudění vzduchu však mohou mít různou složitost podle toho, jak je v nich aditivum patrné a rozlišitelné. Některé záznamy jsou pořízené v laboratorních podmínkách s homogenním pozadím s výrazně odlišnou barvou, než je barva aditiva (tzv. snímky prvního typu). Záznamy, pro jejichž pořízení již nebyly aranžovány vhodné podmínky (tzv. snímky druhého typu), mohou mít méně vhodné pozadí a aditivum již zde není tak patrné. Snímky třetího typu mohou obsahovat velké množství rušivých prvků a zcela nevhodné pozadí, s nímž barva aditiva mnohdy splyne. Práce se tak zabývá i těmito typy snímků.

11.1 VĚDECKÝ PŘÍNOS PRÁCE

První část práce je věnována preprocessingovým úpravám snímku, vhodným pro účely detekce proudnice či rychlosti proudění. Byl vyvinut hranový detektor, pracující na principu nelineární filtrace.

Velká část práce je věnována detektorům využívajícím neuronové sítě. Ty slouží právě k detekci mlhy a pomáhají při určování hranice proudnice. Základní princip je pak doplňován o alternativy a vylepšení, aby výsledný algoritmus dokázal řešit i detekci hranice v nejsložitějších snímcích třetího typu. V rámci této kapitoly byly tyto detektory testovány i na detekci dalších objektů a bylo ověřeno, že jsou velmi vhodné na detekci vzorů, textur a barev. Poslouží tak například k detekci trávy, dřeva či objektů se zřejmou texturou (koberec, stůl, strom...).

V další kapitole je řešena detekce tvaru proudnice vizualizované heliovými bublinami. K sestrojení obecně nekonvexní hranice slouží genetický algoritmus, v rámci něhož se hranice vyvíjí. Dílčím cílem této kapitoly je, aby takto detekovaná hranice byla kvalitativně srovnatelná s hranicemi detekovanými ze snímků proudění mlhy. Postupem bylo dosaženo ohraničení heliových bublin, vhodné pro další zpracování matematickými postupy k určení charakteristik ventilačního systému. Princip algoritmu však není využitelný pouze pro ohraničení proudu heliových bublin, ale lze ho použít i na nekonvexní ohraničení jakékoliv množiny bodů.

Metoda vzájemné korelace, využitelná pro detekci rychlosti proudění, byla doplněna o drobná vylepšení.

Alternativou pro vzájemnou korelaci ve snímcích proudu heliových bublin je metoda sledování částic doplněná o rozhodovací proces, při němž neuronová síť určuje vektor rychlosti v libovolném bodě na základě již známých blízkých vektorů rychlosti.

11.2 PRAKTICKÝ PŘÍNOS PRÁCE

V rámci výzkumu byl vyvíjen komplexní software Interfer, který obsahuje metody zmiňované v této práci. Software byl využíván ve dvou projektech při vývoji ventilačních systému pro nízkoenergetický dům i pro osobní automobil. Software tak umožňuje zpracovávat snímky i záznamy proudění vzduchu obou typů vizualizace. Pomocí něj lze detekovat tvar proudnice i rychlosti proudění a data lze následně vyhodnocovat.

Byla vyvinuta řada metod pro evaluaci tvaru proudnice vizualizované mlhou. Pro jednoduché typy snímků jsou využitelné různé typy prahování či detekce proudnice v řezech. Pro snímky druhého a třetího typu je již vhodnější využít neuronových sítí a komplexnějších postupů zmíněných v práci. Pro určení tvaru proudnice vizualizované heliovými bublinami byl implementován postup tvorby konvexní hranice. Především však byla vyvinuta komplexní metoda pro sestrojení nekonvexní hranice.

Pro určení rychlostí proudění byl implementován postup využívající vzájemné korelace. Pro specifické ne příliš husté proudy částic byla vyvinuta metoda sledování částic a na ni navazující metoda vytvoření vektorové mapy.

Další metody, obsažené ve vyvinutém softwaru, slouží ke statistickému vyhodnocení detekovaných výsledků.

V neposlední řadě je software vhodný k preprocessingovým operacím a k obecným úpravám snímků i sekvencí.

12 SEZNAM POUŽITÉ LITERATURY

- ABRAMOWITZ, M., STEGUN, I.A. Handbook of Mathematical Functions, Applied Mathematics, Series, Volume 55, 1964.
- BASSIOU, N., KOTROPOULOS, C.: *Color image histogram equalization by absolute discounting back-off*, Computer Vision and Image Understanding, vol. 107, no. 1-2, pp.108-122, Jul.-Aug. 2007.
- BRACEWELL, R.: *Pentagram Notation for Cross Correlation*. The Fourier Transform and Its Applications. New York: McGraw-Hill, pp. 46 and 243, 1965.
- CALETKA, P., O. PECH, J. JEDELSKÝ, M. JÍCHA a RICHTER, J.: *Comparison of Methods for Determination of Borders of Stream Visualized by Smoke Method*. In: 34th Conference of Departments of Fluids Mechanics and Thermomechanics Proceedings of Extended Abstracts. Ústí nad Labem: University J. E. Purkyně in Ústí nad Labem. 2015. 11-12. ISBN: 9788074149122.
- CANNY, J., *a Computational Approach To Edge Detection*, IEEE Trans. Pattern Analysis and Machine Intelligence, 8(6):679–698, 1986.
- CERMAK, J. E.: *Wind-tunnel development and trends in applications to civil engineering*, Journal of Wind Engineering and Industrial Aerodynamics, Volume 91, Issue 3, February 2003, Pages 355-370, ISSN 0167-6105.
- DERICHE R., *Using Canny's criteria to derive a recursively implemented optimal edge detector*, Int. J. Computer Vision, Vol. 1, pp. 167–187, April 1987.
- FOJTLÍN, M.; FIŠER, J.: *Tepelný manekýn – možnosti testovania ochrany prostriedkov*, Hazmat protect - sborník konference, pp.1-6, ISBN 978-80-270-0474-4, 2016, SÚJCHBO, v.v.i.
- FREEDMAN, D. A.: *Statistical Models: Theory and Practice*, Cambridge University Press, 2005.
- GALER, M.; HORVAT, L. *Digital Imaging: Essential Skills*. Focal Press, 2003. ISBN 978-0-240-51913-5.
- GONZALES, R. C., WOODS, R.E.: *Digital Image Processing*, Prentice Hall, 2008.
- HAGAN, Martin T. *Neural network design*. USA: PWS, 1996. 734 s. ISBN 7-111-10841-8.
- HARIHARAN, P.: *Basics of Interferometry*. Elsevier Inc. 2007, ISBN 0-12-373589-0.
- HYNEK, J.: *Genetické algoritmy a genetické programování* Grada: Praha 2008
- CHERITION, D., TARJAN, R. E.: *Finding minimum spanning trees*. In: SIAM Journal of Computing, 5 (Dec. 1976), pp. 724–741
- INTERNATIONAL TELECOMMUNICATION UNION. *Norma BT.709-6: Parameter values for the HDTV standards for production and international programme exchange* [online]. ©2015 [cit. 2018-3-22]. Dostupná z: <http://www.itu.int/rec/R-REC-BT.709-6-201506-I>.
- JANOTKOVÁ, E.: *Technika prostředí*. Brno: VUTIUM, 1991, 201 s. ISBN 802140258X.

- JANOTKOVÁ, E.; PAVELEK, M. a kol.: *Použití vizualizačních metod pro sledování proudění vzduchu ve větraných a vytápěných prostorech*, Vytápění, větrání, instalace, Vol.19, 2010, No.5, pp.229-233, ISSN 1210-1389, Společnost pro techniku prostředí článek v časopise
- JARNÍK, V.: *O jistém problému minimálním*, Práce Moravské Přírodovědecké Společnosti, 6, 1930.
- JARVIS, R. A.: *On the identification of the convex hull of a finite set of points in the plane*. Information Processing Letters. 2, 1973, pages 18–21. doi:10.1016/0020-0190(73)90020-3
- JIANG, N., WANG, J., MU, Y.: *Quantum image scaling up based on nearest-neighbor interpolation with integer scaling ratio*, Quantum Information Processing, November 2015, Volume 14, Issue 11, pp 4001–4026.
- KAPINCHEV, K., BRADU, A., BARNES, F., PODOLEANU, A.: *GPU Implementation of Cross-Correlation for Image Generation in Real Time*. 2015. ICSPCS 2015. doi:10.1109/ICSPCS.2015.7391783.
- KEATING, T. J., WOLF, P. R., SCARPACE, F. L.: *An Improved Method of Digital Image Correlation*, Photogrammetric Engineering and Remote Sensing 41(8): 993–1002, 1975.
- KHARE, C., NAGWANSHI, K. K.: *Image Restoration Technique with Non Linear Filter*, International Journal of Advanced Science and Technology ,Vol. 39, February, 2012.
- KOZA, J. R.: *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. The MIT Press, 1992, ISBN 0-262-11170-5.
- LÍZAL, F.; JEDELSKÝ, J.; JÍCHA, M.; HAPALA, R.: *Rychlostní pole v okolí vyústky pro odmrzování a odmlžování čelního skla osobního automobilu*, Vytápění, větrání, instalace, Vol.25, 2016, No.4, pp.190-196, ISSN 1210-1389, Society of Environmental Engineering.
- LÝSEK, J., ŠTASTNÝ, J., MOTYČKA, A. *Object Recognition by Means of Evolved Detector and Classifier Program*. In MENDEL 2012, 18th International Conference on Soft Computing. Brno University of Technology, 2012, p. 82-87. ISBN 978-80-214-4540-6.
- MCCULLOCH, W., PITTS, W.: *a logical calculus of the ideas immanent in nervous activity*. Bulletin of Mathematical Biophysics, 5:115–133, 1943.
- MERZKIRCH, W.: *Flow visualization*. New York: Academic Press, 1987. ISBN 0-12-491351-2.
- MITCHELL, M.: *An Introduction to Genetic Algorithms*. Cambridge, MA: MIT Press 1996.
- PAVELEK, M., JANOTKOVÁ, E., ŠTĚTINA, J.: *Visualization and optical measuring methods*. FSI VUT, Brno 2001
- POKORNÝ, J.; POLÁČEK, F.; FOJTLÍN, M.; FIŠER, J.; JÍCHA, M.: *Measurement of airflow and pressure characteristics of a fan built in a car ventilation system*, EPJ Web of Conferences, Vol.114, (2016), No.2016, pp.644-647, ISSN 2100-014X, EDP Sciences - Experimental fluid mechanics 2015, Praha, 17.11.2015-20.11.2015.
- RAFFEL, M. a kol: *Particle Image Velocimetry* Springer 2007, ISBN 978-3-540-72307-3 Second Edition Springer Berlin Heidelberg New York.

- RICHTER, J., ŠŤASTNÝ, J., *Evaluation of centres of stripes in the interferogram image*. In International Conference AICT 2011, Recent Research in Applied Informatics. Prague, Czech republic, 2011. pp. 125-130. ISBN 978-1-61804-034-3.
- RICHTER, J., ŠŤASTNÝ, J.: *Fog Border Detection in the Airflow Image*. In Advances in Computer Science. Proceedings of the 6th WSEAS European Computing Conference ECC'12. Prague, Czech Republic 2012, pp. 134-139, ISBN 978-1-61804-126-5.
- RICHTER, J., ŠŤASTNÝ, J., JEDELSKÝ, J. *Estimations of Shape and Direction of an Air Jet Using Neural Networks*. In MENDEL 2013, 19th International Conference on Soft Computing. Brno University of Technology, 2013, p. 221-226. ISBN 978-80-214-4755-4.
- SUTTON, M. A., ORTEU, J. J., SCHREIER, H.: *Image Correlation for Shape, Motion and Deformation Measurements*, Hardcover, 2009, ISBN 978-0-387-78746-6.
- ŠKORPIL, V., RICHTER, J., ŠŤASTNÝ, J., LÝSEK, J., SZÖCZ, J.: *Computer Analysis of Airflow Images Visualized by Smoke*. In International Conference CSCC 2012, Recent Researches in Communication and Computers. Kos Island, Greece, 2012. pp. 89-94. ISBN 978-1-61804-109-8.
- ŠŤASTNÝ, J., RICHTER, J. *Using neural networks for determining velocity vectors of air flow visualized by helium bubbles*. Acta Universitatis Agriculturae et Silviculturae Mendelianae Brunensis, 2014, vol. 62, issue 4, pages 757-768, ISSN 1211-8516, (index. Scopus).
- ŠŤASTNÝ, J., RICHTER, J.: *Adaptation of genetic algorithm to determine air jet shape of flow visualized by helium bubbles*. Journal of Aerospace Engineering, 2016, vol. 29, issue 6, pages 12, ISSN 0893-1321, (index. WoS: IF 1.107, Scopus).
- ŠŤASTNÝ, J., ŠKORPIL, V.: *Genetic Algorithm and Neural Network*. In Proceedings of the 7th WSEAS International Conference on Applied Informatics and Communications, Vouliagmeni, Greece, 2007, p. 347-351. ISSN 1790-5117. ISBN 978-960-8457-96-6.
- ZHOU, P., YE, W., WANG, Q.: *An Improved Canny Algorithm for Edge Detection*. Journal of Computational Information Systems, 2011

13 SEZNAM POUŽITÝCH ZKRATEK A SYMBOLŮ

<i>CMY</i>	barevný model (azurová, purpurová, žlutá)
<i>CMYK</i>	barevný model (azurová, purpurová, žlutá, černá)
<i>GA</i>	Genetický algoritmus
<i>HSV</i>	barevný model (odstín, sytost, jas)
<i>MLP</i>	multilayer perceptron network = vícevrstvá perceptronová síť
<i>NHD</i>	Nelineární hranový detektor
<i>RGB</i>	barevný model (červená, zelená, modrá)
<i>RGBA</i>	barevný model (červená, zelená, modrá, alfa)

14 SEZNAM OBRÁZKŮ

Snímek 1	Příklad snímku prvního typu Zdroj: Odbor termomechaniky a techniky prostředí, VUT FSI [s.a.]	17
Snímek 2	Příklad snímku druhého typu Zdroj: Odbor termomechaniky a techniky prostředí, VUT FSI [s.a.]	17
Snímek 3	Příklad snímku třetího typu Zdroj: Odbor termomechaniky a techniky prostředí, VUT FSI [s.a.]	18
Snímek 4	Schéma struktury neuronu navržená McCullochem a Pittsem (1943)	23
Snímek 5	Příklad struktury <i>MLP</i> 3-5-2	24
Snímek 6	Obrázek pro detekci hran Zdroj: Auto.cz, 2013	30
Snímek 7	Výsledek nelineární hranové detekce bez prahování, $k_v = 2$	31
Snímek 8	Výsledek hranové detekce Cannyho hranovým detektorem	31
Snímek 9	Hranová detekce neuronovou sítí	33
Snímek 10	Podmíněné prahování Zdroj: Odbor termomechaniky a techniky prostředí, VUT FSI [s.a.]	36
Snímek 11	Exemplární snímek se znázorněnou křivkou řezu Zdroj: Odbor termomechaniky a techniky prostředí, VUT FSI [s.a.]	38
Snímek 12	Vyhodnocení řezu ve snímku 11	39
Snímek 13	Detekce paty v průběhu jasu.....	41
Snímek 14	Hranice sestavená vyhodnocením po sobě jdoucích řezů.....	42
Snímek 15	Předání vzorů neuronové sítí	44
Snímek 16	Vyhodnocení snímku 1. typu neuronovou sítí.....	44
Snímek 17	Vyhodnocení snímku 3. typu neuronovou sítí.....	45
Snímek 18	Detekovaná mlha po aplikaci vyhlazovacího algoritmu	46
Snímek 19	Detekce mlhy sítí, naučenou na snímku 17a, v podobných snímcích Zdroj: Odbor termomechaniky a techniky prostředí, VUT FSI [s.a.]	47
Snímek 20	Porovnání oblastí podobné barvy Zdroj: Richter, Šťastný, Jedelský, 2013	47
Snímek 21	Porovnání základní a spojitě detekce mlhy neuronovou sítí	49
Snímek 22	Porovnání výsledků různých neuronových sítí	50
Snímek 23	Vyhodnocení obecným expertem	51
Snímek 24	Vyhodnocení specifickým expertem	51
Snímek 25	Spojité detekce	52
Snímek 26	Vyhodnocení dvojicí expertů před sloučením.....	53
Snímek 27	Výsledek sloučení jednotlivých prahovaných snímků	53

Snímek 28	Snímek sloučený z dílčích vyhodnocení oběma experty.....	54
Snímek 29	Výsledek podmíněného prahování sloučeného výsledku.....	55
Snímek 30	Detekce koberce	55
Snímek 31	Detekce červených krovek Berušky Zdroj: rajce.net, 2014	56
Snímek 32	Významné hodnoty modré barevné složky	58
Snímek 33	Prahování k označení pixelů heliových bublin Zdroj: Odbor termomechaniky a techniky prostředí, VUT FSI [s.a.]	59
Snímek 34	Shluk dvou bublin – detail.....	60
Snímek 35	Výběr reprezentantů bublin v okolí kolem vyústky	61
Snímek 36	Minimální kostra grafu reprezentantů shluků heliových bublin Zdroj: Šťastný, Richter, 2016	62
Snímek 37	Reprezentanti R_i seřazení podle vzdálenosti jim příslušících přímek q_i (kolmých na hlavní směr proudění) od počátku proudění	63
Snímek 38	Nekonvexní obálka.....	64
Snímek 39	Princip tvorby konvexní obálky	64
Snímek 40	Konvexní obálka reprezentantů shluků heliových bublin	65
Snímek 41	Porovnání konvexní a nekonvexní hranice Zdroj: Šťastný, Richter, 2016	65
Snímek 42	Sestrojená hranice nad danou množinou bodů pro chromozom 1101010 Zdroj: Šťastný, Richter 2016	69
Snímek 43	Volba konce pro připojení vyšetřovaného bodu.....	70
Snímek 44	Průnik hranice při rozhodovací geometrické konstrukci.....	70
Snímek 45	Hledání bližšího konce ve směru kolmém na směr proudění.....	71
Snímek 46	Příklad hranice s návratovou úsečkou (zelená)	72
Snímek 47	Hledání cesty z bodu na hranici	73
Snímek 48	Rozšíření čtvercové obálky	73
Snímek 49	Výsledek genetického algoritmu pro 20 bodů.....	75
Snímek 50	Výsledek genetického algoritmu pro 40 bodů.....	75
Snímek 51	Výsledek genetického algoritmu pro 60 bodů.....	76
Snímek 52	Výsledek genetického algoritmu pro 120 bodů.....	76
Snímek 53	Výsledek genetického algoritmu pro reálný snímek	77
Snímek 54	Příklad vektorové mapy.....	81
Snímek 55	Ilustrace generování cest pro bod R^0_0 snímku S^0 pro $u = v = 3$	85
Snímek 56	Ideální cesta, chybně nalezená z důvodu chybějícího faktoru vzdálenosti následovníka	86

Snímek 57	Výsledek detekce vektorů rychlosti	88
Snímek 58	Vymazání nesprávného vektoru dle okolí	90
Snímek 59	Výběr bublin pro učení dotyčných vektorů	91
Snímek 60	Bublina uvnitř trojúhelníku	92
Snímek 61	Transformace vstupu	94
Snímek 62	Vektorová mapa dle postupu sledování částic vytvořená neuronovou sítí	96
Snímek 63	Hledání nejdelší úsečky v ohraničené oblasti – nalezena (P_2, P_3)	98
Snímek 64	Hledání nejmenší kruhové výseče s hranicí p_1 danou sklonem 0, který obsahuje všechny detekované přímky.	101
Snímek 65	Nespojitá hranice po sčítání pixelů.....	102
Snímek 66	Lineární filtrace snímku 62	103
Snímek 67	Prahování snímku 63 s prahem 40	103
Snímek 68	Určení tvaru proudnice metodou podmíněné detekce dvojicí expertů.....	107
Snímek 69	Nekonvexní ohraničení bodů s využitím genetických algoritmů (50 bodů)	108
Snímek 70	Vektorová mapa vytvořená použitím vzájemné korelace Zdroj: Šťastný, Richter, 2014	109
Snímek 71	Vektorová mapa vytvořená metodou sledování částic Zdroj: Šťastný, Richter, 2014	109

15 SEZNAM TABULEK

Tabulka 1	Křížení náhodně dlouhými úseky chromozomu.....	27
Tabulka 2	Křížení náhodnými rodiči pro každý gen	27
Tabulka 3	Příklad pro demonstraci vyhlazovacího algoritmu.....	37
Tabulka 4	Doporučená velikost populace pro daný počet reprezentantů.....	66
Tabulka 5	Pravděpodobnost počtu mutací v závislosti na počtu genů.....	67
Tabulka 6	Příklad množiny reprezentantů s příslušnými geny	68
Tabulka 7	Správně seřazené hraniční body (požadovaný výstup)	68
Tabulka 8	Výsledky testování genetického algoritmu	77

16 SEZNAM PŘÍLOH

Na přiloženém CD jsou dvě aplikace:

HEGenetic.exe

Jedná se o testovací prostředí pro metodu detekce nekonvexního ohraničení bodů pomocí genetických algoritmů. Náhodně generuje množinu definovaného počtu bodů a spouští genetický algoritmus. Souhrnná funkce pak pro zvolený počet opakování vykonává celý proces (vygenerování bodů, spuštění genetického algoritmu, uložení).

Interfer.exe

Jde o komplexní prostředí sloužící pro práci s obrázky, složkami snímků i záznamy. Obsahuje množství preprocessingových funkcí, úprav obrazu i statistických metod. V neposlední řadě obsahuje veškeré funkce k vyhodnocení snímků proudění vzduchu, zmíněné v této práci.

Každá úprava se provádí v konkrétní oblasti (celý snímek, vytvořená oblast, detekovaná oblast) a je zaznamenávána, aby postup úprav mohl být replikován na jiné snímky (například další snímky sekvence).

Manual_Interfer.pdf

CD obsahuje kromě vlastního softwaru Interfer.exe také návod na jeho použití.